

Manual Rápido



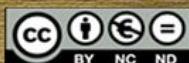
Centro Don Bosco
Villamuriel de Cerrato

Sublime Text

2

Apuntes realizados para el ciclo de FP Grado Superior:
Administración de Sistemas Informáticos en Red

Autor: Jorge Sánchez Asenjo (www.jorgesanchez.net)
Versión del documento: 1.6, Año 2012



Esta obra está bajo una licencia de Reconocimiento-NoComercial-CompartirIgual de Creative Commons
Para ver una copia de esta licencia, visite: <http://creativecommons.org/licenses/by-nc-sa/3.0/deed.es>



Atribución-NoComercial-CompartirIgual 3.0 Unported (CC BY-NC-SA 3.0)

Esto es un resumen fácilmente legible del [Texto Legal \(la licencia completa\)](http://creativecommons.org/licenses/by-nc-sa/3.0/legalcode).

<http://creativecommons.org/licenses/by-nc-sa/3.0/legalcode>

Usted es libre de:

Compartir - copiar, distribuir, ejecutar y comunicar públicamente la obra
hacer obras derivadas

Bajo las condiciones siguientes:



Atribución — Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciante (pero no de una manera que sugiera que tiene su apoyo o que apoyan el uso que hace de su obra).



No Comercial — No puede utilizar esta obra para fines comerciales.



Compartir bajo la Misma Licencia — Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

Entendiendo que:

Renuncia — Alguna de estas condiciones puede **no aplicarse** si se obtiene el permiso del titular de los derechos de autor

Dominio Público — Cuando la obra o alguno de sus elementos se halle en el **dominio público** según la ley vigente aplicable, esta situación no quedará afectada por la licencia.

Otros derechos — Los derechos siguientes no quedan afectados por la licencia de ninguna manera:

- Los derechos derivados de **usos legítimos** u otras limitaciones reconocidas por ley no se ven afectados por lo anterior.
- Los derechos **morales** del autor;
- Derechos que pueden ostentar otras personas sobre la propia obra o su uso, como por ejemplo **derechos de imagen** o de privacidad.

índice

(1) ¿Qué es Sublime Text 2	7
(1.1) editor de texto multipropósito	7
(1.2) ventajas sobre sus competidores	7
(1.3) desventajas de Sublime Text	7
(2) instalación	8
(2.1) instalación en entorno Windows	8
(2.1.1) posibilidades	8
(2.1.2) instalación ZIP	8
(3) manejo básico de Sublime Text	11
(3.1) escritura de código	11
(3.2) manejo de documentos	12
(3.3) opciones visuales	14
(3.4) edición de documentos	16
(3.4.1) multicursor	16
(3.4.2) teclas de selección de texto	17
(3.4.3) teclas de edición de texto	18
(3.4.4) contraer código	19
(3.4.5) marcas	20
(3.4.6) teclas y acciones de edición de etiquetas (HTML, XML)	21
(3.4.7) uso de marcadores (bookmarks)	21
(3.5) búsqueda	21
(3.5.1) cuadro de búsqueda	21
(3.5.2) opciones avanzadas del cuadro de búsqueda	23
(3.5.3) reemplazar	23
(3.5.4) buscar en varios archivos	23
(3.6) ir a cualquier parte	23
(3.6.1) abrir documentos rápidamente	23
(3.6.2) ir a números de línea	24
(3.6.3) buscar texto	24
(3.6.4) buscar métodos	24
(3.7) trabajar con proyectos	24
(3.7.1) guardar el proyecto	24
(3.7.2) cerrar el proyecto	24
(3.7.3) abrir rápidamente un proyecto	25
(4) instalación de paquetes	26
(4.1.1) Package Control	26

(4.1.2) instalar paquetes	28
(4.1.3) paquetes interesantes (especialmente para PHP y XML/HTML)	28
(5) personalizar Sublime Text	35
(5.1) ajustes y teclas	35
(5.1.1) personalizar ajustes	35
(5.1.2) personalizar teclas	36
(5.1.3) personalizar ajustes en paquetes	36
(5.2) personalizar colores y temas	36
(5.2.1) personalizar temas	36
(5.2.2) personalizar colores de otros paquetes	37
(5.3) macros	37
(5.4) snippets	37
(5.4.2) introducir códigos de cursor	40
(5.4.3) snippets para ZenCoding	40
apéndice: modificación del Path del Sistema	45

(1) ¿Qué es Sublime Text 2

(1.1) editor de texto multipropósito

Sublime Text 2 es un editor de texto pensado para escribir código en la mayoría de lenguajes de programación y formatos documentales de texto, utilizados en la actualidad : Java, Python, Perl, HTML, JavaScript, CSS, HTML, XML, PHP, C, C++, etc., etc.

Es un editor de código que vale 59\$, pero que permite un uso ilimitado en el tiempo de manera gratuita (si no se va a hacer un uso comercial del software). De vez en cuando una ventanita aparece en el programa para animarnos a registrar y pagar el programa (indudablemente vale ese precio, pero podemos probarlo todo el tiempo que queramos hasta decidir comprarlo, o no).

Permite escribir todo tipo de documentos de código en formato de texto y es capaz de colorear el código, ayudarnos a la escritura, corregir mientras escribimos, usar abreviaturas (snippets), ampliar sus posibilidades, personalizar hasta el último detalles,... casi cualquier cosa que le podamos pedir a un editor.

(1.2) ventajas sobre sus competidores

- Es un programa **muy rápido** en su ejecución. Todo en él funciona de manera extremadamente veloz.
- Es **muy ligero**. Ocupa apenas siete megabytes, por lo que no consume apenas recursos en el ordenador. Lo que le hace una opción muy interesante frente a entornos integrados de codificación con grandes herramientas (como **Eclipse** o **NetBeans**), pero que resultan extremadamente pesados en su ejecución.
- Permite codificar en **casi cualquier lenguaje**.
- Tiene **gran cantidad de paquetes** que mejoran enormemente sus prestaciones.
- **Permite configurar cada aspecto** casi del programa y adaptarle absolutamente a nuestras necesidades
- Es **multiplataforma**. Funciona tanto en Windows como en Linux como en entorno Mac.
- Tiene todas las posibilidades de ayuda al codificar que se le pueden pedir a un editor.
- Su crecimiento está resultando exponencial, por lo que posee una comunidad de usuarios cada vez mayor.
- Tiene posibilidades incluso de depurar y ejecutar el código sin salir del editor; así como opciones de gestión de proyectos completos de trabajo.

(1.3) desventajas de Sublime Text

- La fundamental es que es difícil de aprender y configurar al principio al ser un editor de texto con filosofía de editor clásico (como **vim**), lo que puede resultar dificultoso para usuarios acostumbrados a herramientas más visuales o a aquellas personas que están empezando en el mundo del desarrollo de aplicaciones o páginas web.
- Aún posee algunos fallos, aunque no mayores que otros productos más veteranos.

(2) instalación

(2.1) instalación en entorno Windows

(2.1.1) posibilidades

En el caso de Windows hay dos formas de instalar:

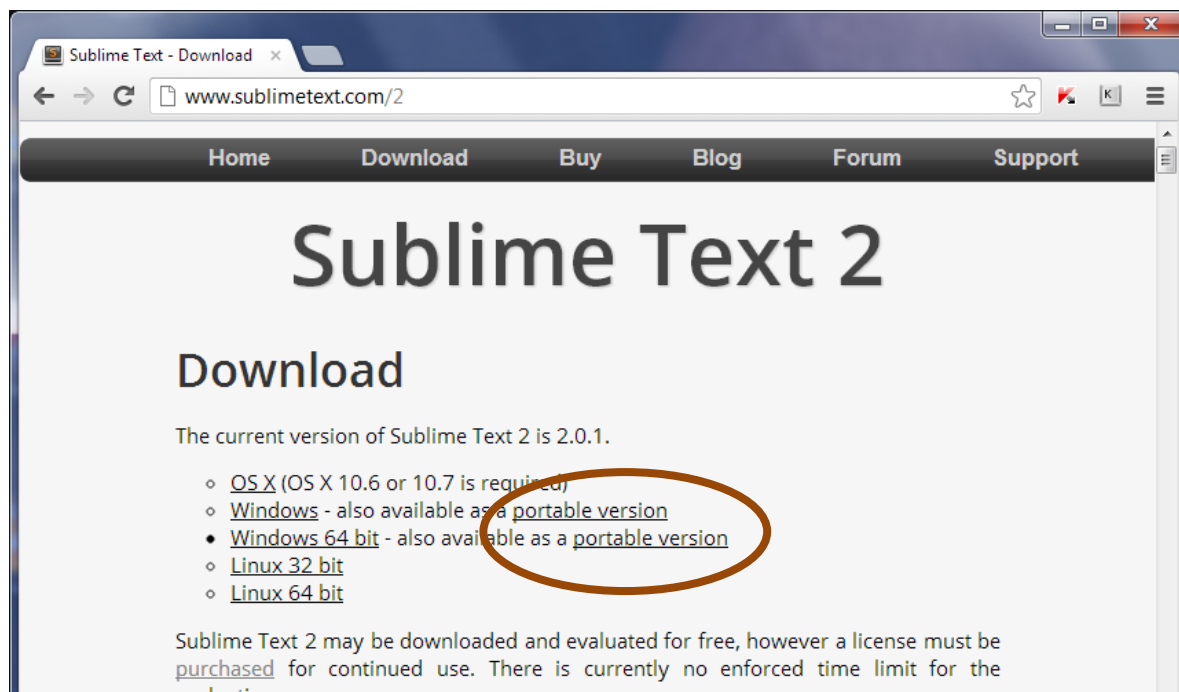
- (1) **Instalación como ejecutable.** En ese caso se instala como un programa más dentro de las aplicaciones de Windows. La ventaja es que funciona de la forma en la que los usuarios de Windows están más acostumbrados, la desventaja es que sólo se instala para un equipo.
- (2) **Instalación como paquete ZIP, instalación portable.** En este caso se trata de un archivo ZIP que se descarga de Internet y se descomprime en una ubicación que deseemos. La ventaja es que la carpeta en la que hemos metido la instalación la podemos copiar a otro equipo y el programa funcionará perfectamente; es más, podemos copiar la carpeta en un pincho y así llevarnos la instalación siempre con nosotros. Ésta es la más aconsejable.

En el caso de Linux se puede descargar un archivo **tar**, pero está disponible como paquete en la mayoría de repositorios de los sistemas Linux.

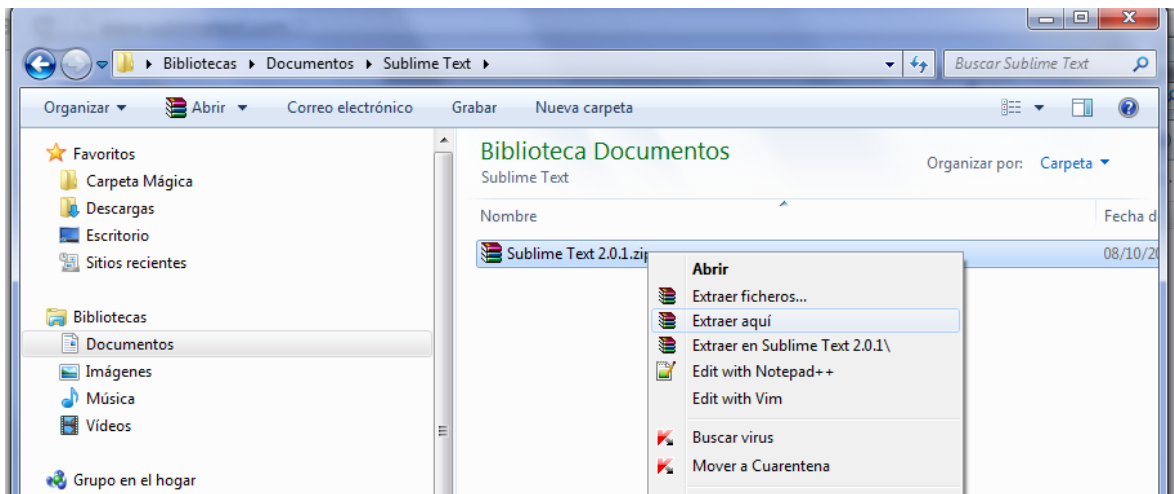
(2.1.2) instalación ZIP

Los pasos para instalar Sublime Text2 de forma portable en Windows son:

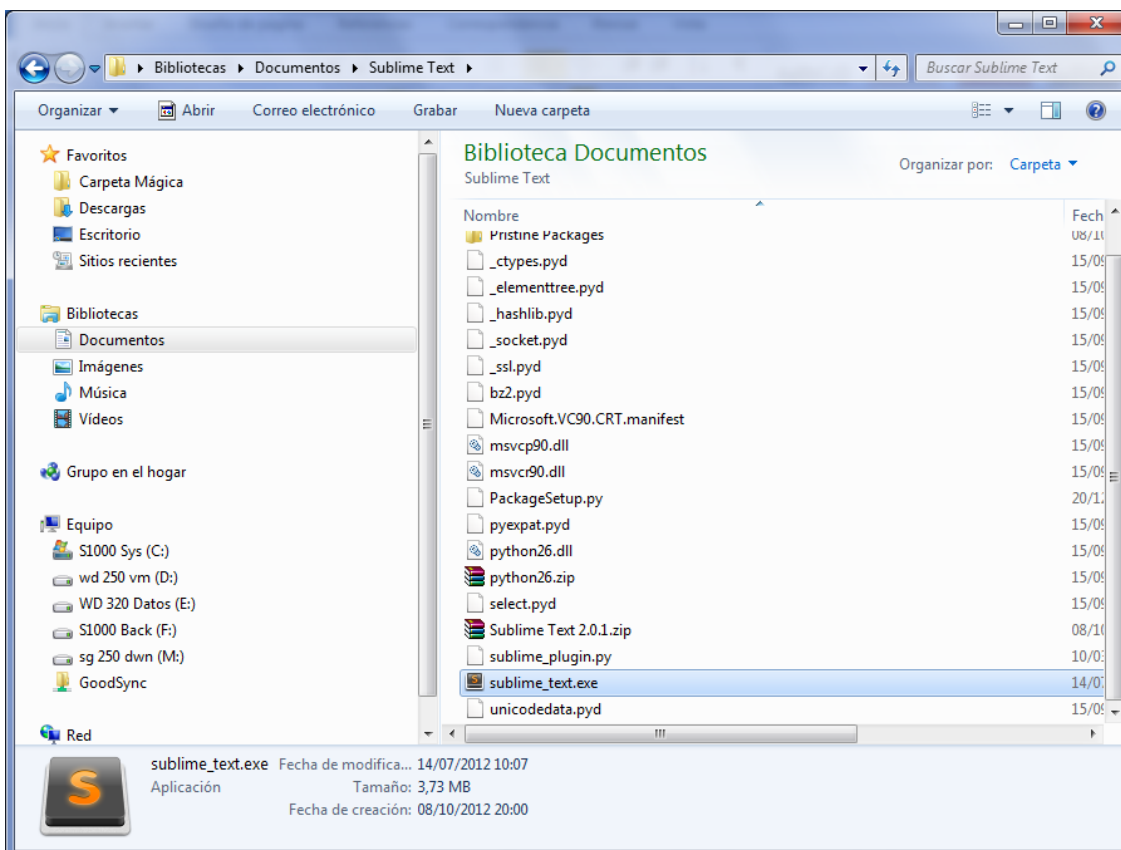
- (1) Ir a la página www.sublimetext.com/2



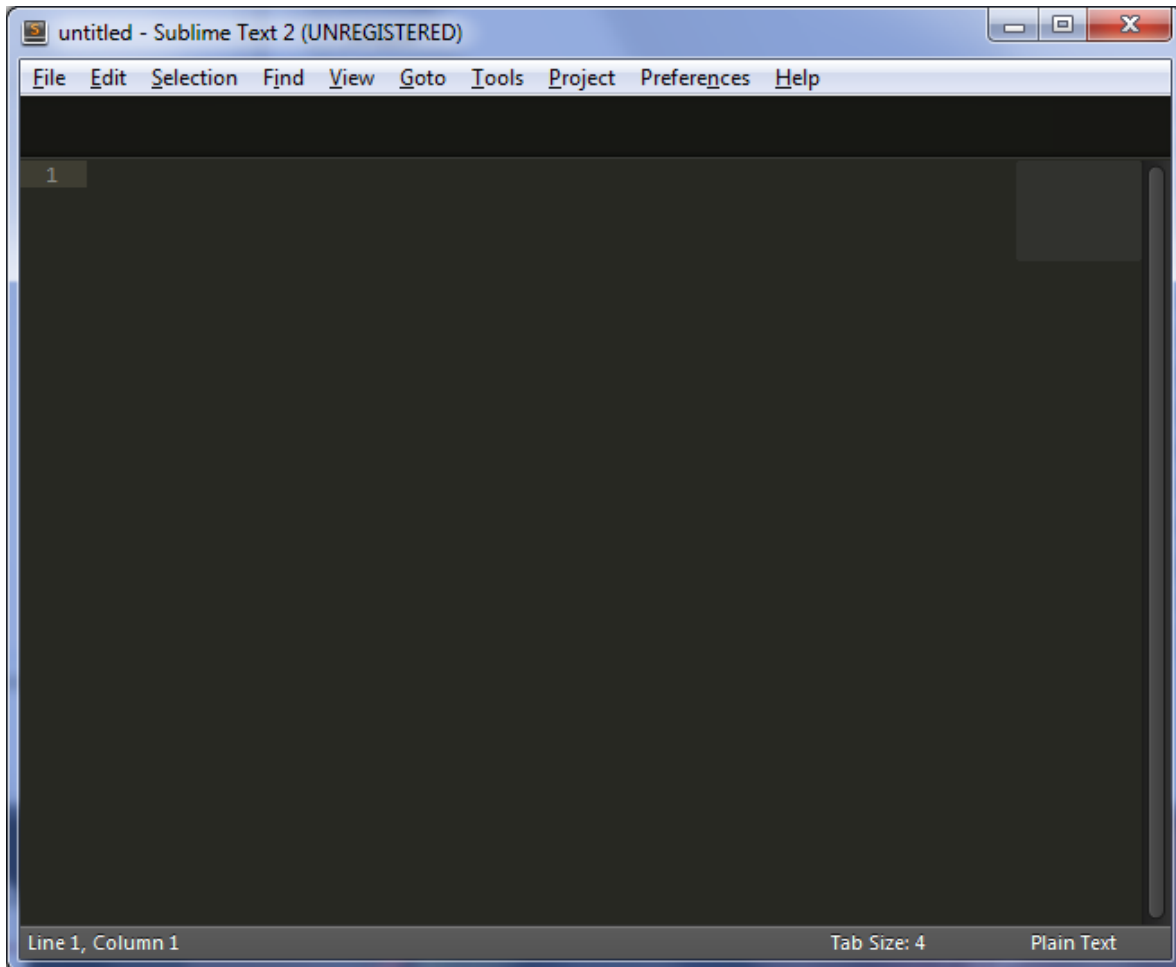
- (2) Hacer clic para descargar el archivo versión portable del programa (estará en formato ZIP)
- (3) El archivo descargado se ha de copiar en la carpeta en la que deseamos instalar el programa. Puede ser cualquier carpeta que creemos (por ejemplo **Sublime Text en Mis documentos**).
- (4) Pulsar el botón derecho en el archivo descargado y elegir **Extraer aquí**.



- (5) Una vez extraído, aparecen numerosos archivos. EL programa arranca mediante el archivo **sublime_text**



- (6) *(opcional)* Para mayor comodidad podemos hacer un acceso directo al escritorio a ese archivo para que cada vez que trabajemos con el programa sea más cómodo. Para ello basta con pulsar el botón derecho sobre el archivo `sublime_text` (icono con la S de color rojo), elegir **copiar**. Luego ir al escritorio y elegir **pegar acceso directo** tras pulsar el botón derecho en el escritorio.
- (7) Hacer doble clic sobre el acceso al programa y aparecerá la pantalla de Sublime Text inicial:



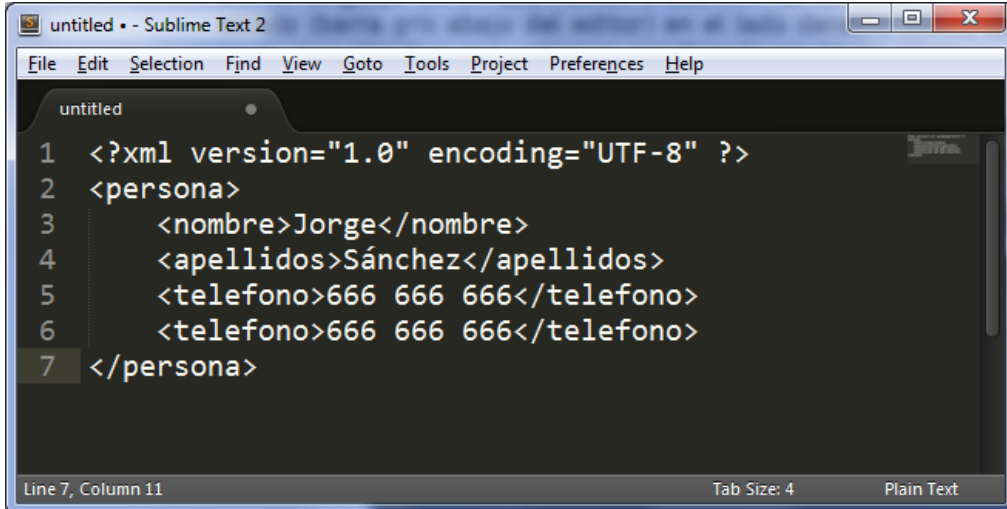
- (8) La palabra **UNREGISTERED** en la barra de títulos hace referencia a que no hemos pagado el programa y que sería conveniente hacerlo
- (9) En cuanto la hagamos dejará de aparecer (a cambio de 59 dólares) y con ello los avisos animando a pagar.

(3) manejo básico de Sublime Text

(3.1) escritura de código

Sublime Text sirve para escribir código, por ello debemos avisar al programa cuál va a ser el lenguaje en el que vamos a escribir. Es se hace haciendo clic en la barra de estado (barra gris abajo del editor) en el lado derecho donde aparecerá el texto *Plain text*, que significa que estamos escribiendo en texto plano y no en código de algún lenguaje.

Si no lo hacemos, el programa no ayuda a escribir y así si escribimos un código XML sin indicar nada (aunque el programa a veces lo adivina) aparecería tal que así:

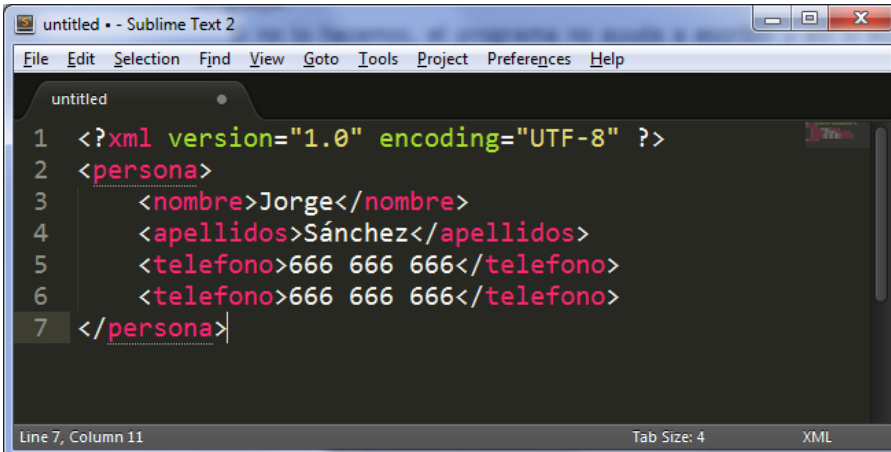


The screenshot shows the Sublime Text 2 interface with a file named 'untitled'. The code is as follows:

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <persona>
3     <nombre>Jorge</nombre>
4     <apellidos>Sánchez</apellidos>
5     <telefono>666 666 666</telefono>
6     <telefono>666 666 666</telefono>
7 </persona>
```

The status bar at the bottom indicates 'Line 7, Column 11', 'Tab Size: 4', and 'Plain Text'.

Tras avisar de que estamos escribiendo XML:



The screenshot shows the same Sublime Text 2 interface, but now the code is syntax-highlighted. The status bar at the bottom indicates 'Line 7, Column 11', 'Tab Size: 4', and 'XML'.

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <persona>
3     <nombre>Jorge</nombre>
4     <apellidos>Sánchez</apellidos>
5     <telefono>666 666 666</telefono>
6     <telefono>666 666 666</telefono>
7 </persona>
```

El código se lee mejor

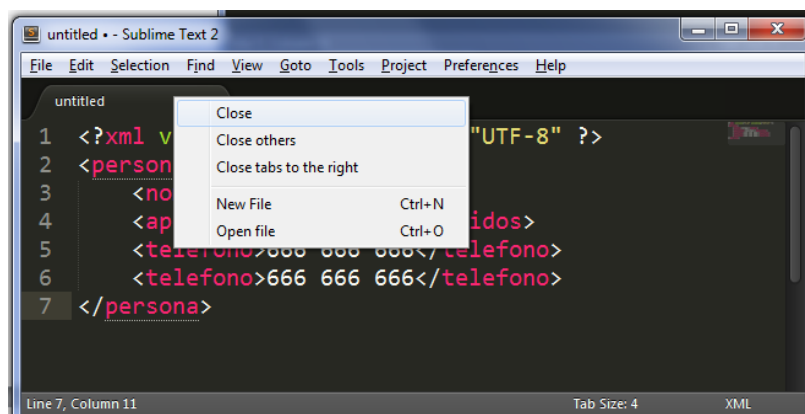
Sublime Text entiende código de muchísimos lenguajes, basta con indicar el lenguaje que estamos usando, o guardar el documento con la extensión apropiada.

La tecla **Ctrl+Espacio** nos intenta ayudar al escribir al pulsarla una pequeña ventana nos intenta sugerir palabras para continuar. A medida que personalizemos el entorno, esta tecla será más potente

(3.2) manejo de documentos

Las siguientes acciones están relacionadas con los documentos. Las teclas es interesante aprenderlas porque nos permiten trabajar más rápido. Lo aconsejable es fijarnos en qué operaciones hacemos a menudo y forzarse a aprender la tecla, a medio plazo ganaremos muchísimo tiempo:

- **File-New o tecla Ctrl+N.** Permite crear un nuevo documento. Simplemente aparece una nueva pestaña donde podremos escribir las cosas del nuevo documento.
- **File-Open o tecla Ctrl+O.** Permite abrir un documento previamente guardado. También podemos directamente arrastrar el documento hacia el programa Sublime Text directamente.
- **File-Save o Ctrl+S.** Guarda el documento actual. Si es la primera vez un cuadro nos permite elegir dónde guardamos el documento y el nombre del mismo (conviene escribir la extensión al poner el nombre, por ejemplo [documento.xml](#))
- **File-Save with encoding.** Igual que la anterior, pero podemos elegir la forma de codificar los caracteres del documento (por ejemplo UTF-8). De otro modo dependemos de la elección de codificación que haga el programa por defecto
- **File-Save as. Guardar como,** permite guardar el documento con otro nombre o en otra carpeta (o ambas cosas)
- **File-Save all.** Permite guardar todos los documentos que tengamos abiertos.
- **Botón derecho en las pestañas-Close, o tecla Ctrl+W.** Cierra el documento a cuya pestaña apuntamos



- **Botón derecho en las pestañas-Close others.** Cierra todos los documentos menos aquel al que estamos apuntando
- **Botón derecho en las pestañas-Close tabs to the right.** Cierra los documentos que se encuentran a la derecha del actual.
- **File-Close all files.** Cierra todos los documentos abiertos
- **File-New View Into File.** Permite abrir otra pestaña para el documento actual para así poder disponer de dos vistas del documento en distintas pestañas.
- **File-New Folder.** Permite abrir una carpeta completa. De ese modo aparecerá la barra lateral de archivos (sino, basta con elegir **View-Show Side Bar** en el menú) y con esa barra podemos gestionar los archivos de la carpeta. Pulsando el botón derecho en el sitio apropiado de la carpeta podremos elegir:

- **New File.** Para crear un nuevo archivo
- **New Folder.** Para crear otra carpeta
- **Rename.** Para cambiar el nombre
- **Delete Folder o Delete File.** Para eliminar el archivo o la carpeta sobre la que se pulsó el botón derecho.

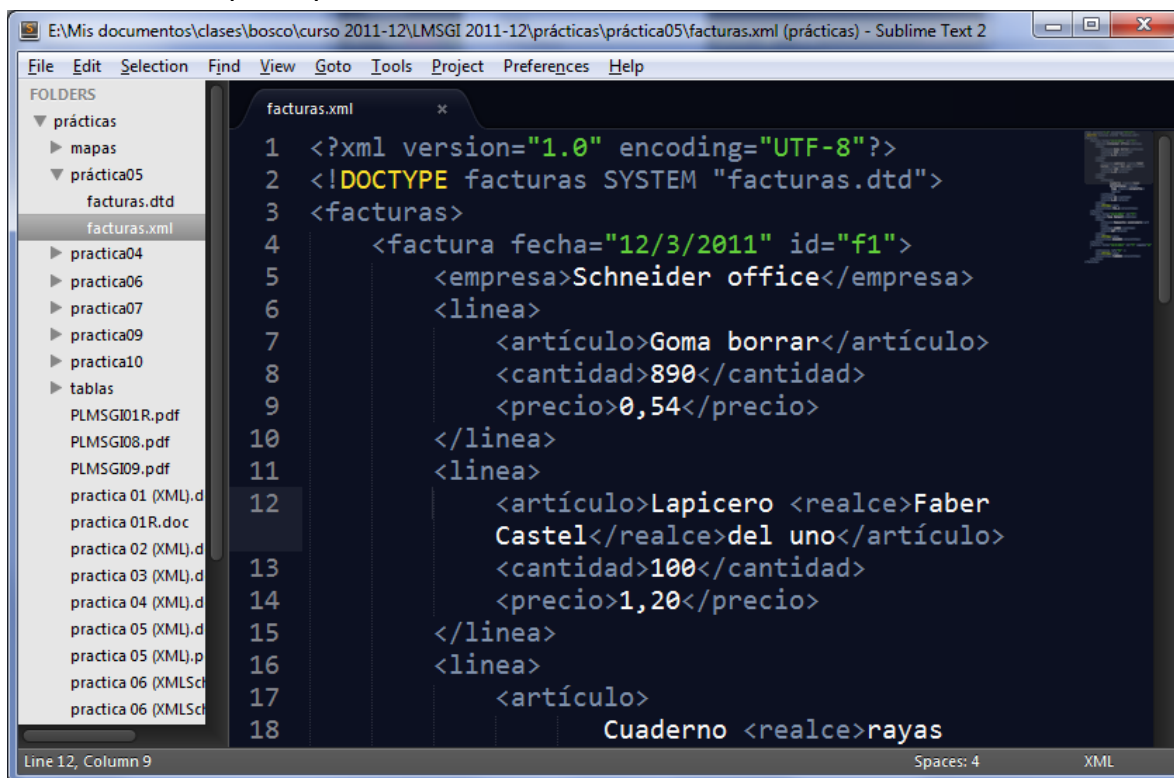
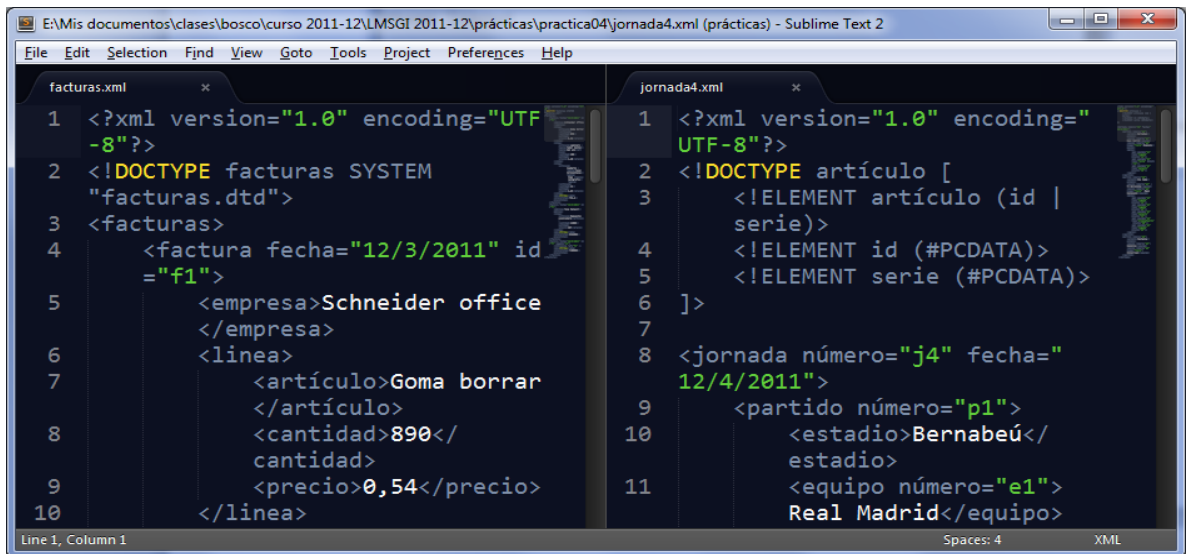


Ilustración 1, Sublime Text tras abrir una carpeta con File-Open Folder

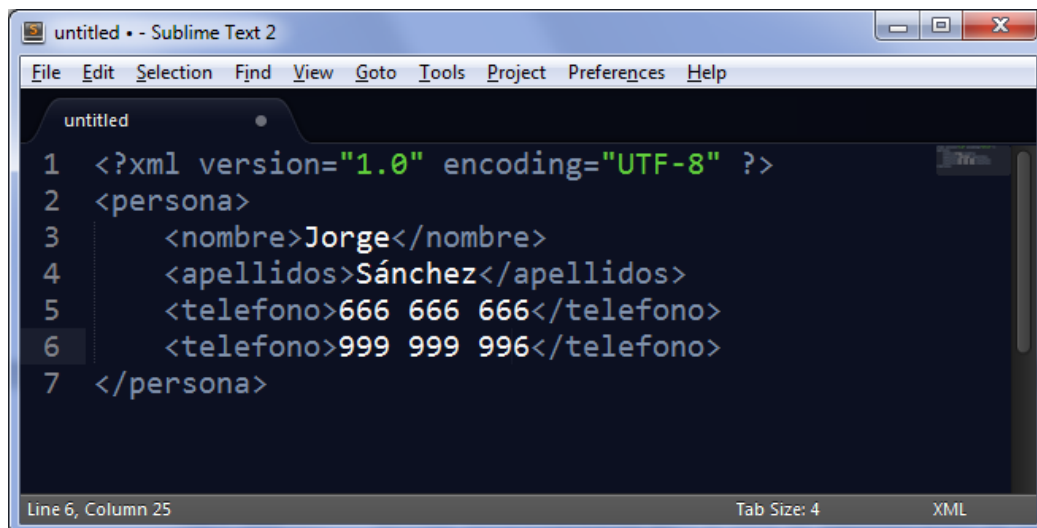
- **Alt+1 o Alt+2, Alt+3....** Con esas teclas vamos rápidamente al primer documento abierto o al segundo o al tercero o al...
- **Ctrl+RePág o Ctrl+AvPág.** Va a la pestaña anterior o a la siguiente respectivamente.
- **View Layout.** Permite cambiar la manera de mostrar documentos del programa. También se puede con las teclas **Alt+Shift(Mayús)** seguidas de un número. Por ejemplo **Alt+Mayús+2** muestra el programa a dos columnas y así podremos abrir un documento en la primera y otro en la segunda:



- **Ctrl+Mayús+1, Ctrl+Mayús+2,...** Cuando hemos dividido la pantalla en filas o en columnas, permite pasar el documento abierto al grupo indicado por la tecla. Por ejemplo si estamos a dos columnas y elegimos un archivo de la primera columna y pulsamos Ctrl+Mayús+2, éste se abrirá en la segunda columna. También podemos abrir un documento y directamente arrastrarle a la columna donde más nos interese ponerle

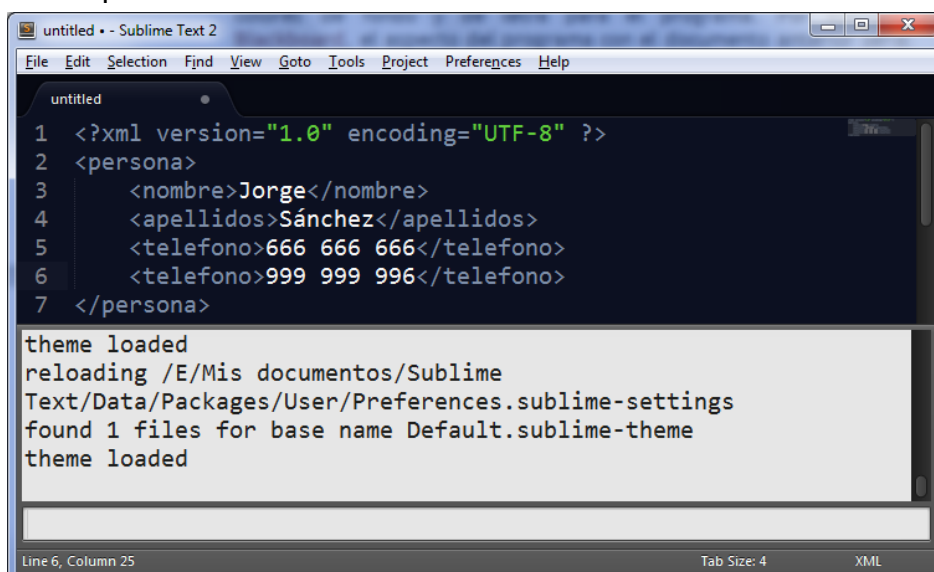
(3.3) opciones visuales

- **Preferences-Color Scheme.** Permite elegir otro tema visual, es decir otros colores de fondo y de letra para el programa. Por ejemplo eligiendo **Blackboard**, el aspecto del programa sería:



- **Preferences-Font.** Permite modificar el tamaño de la letra. Las teclas **Ctrl+signo más (Ctrl+'+') y Ctrl+signo menos (Ctrl+'-')** hacen también lo mismo de forma mucho más veloz
- **Preferences-Font-Reset.** Deja la letra al tamaño original

- **View-Show Console ó Ctrl+ñ.** Muestra la consola, una ventana desde la que se pueden introducir comandos Python al programa. También informa de las últimas operaciones:

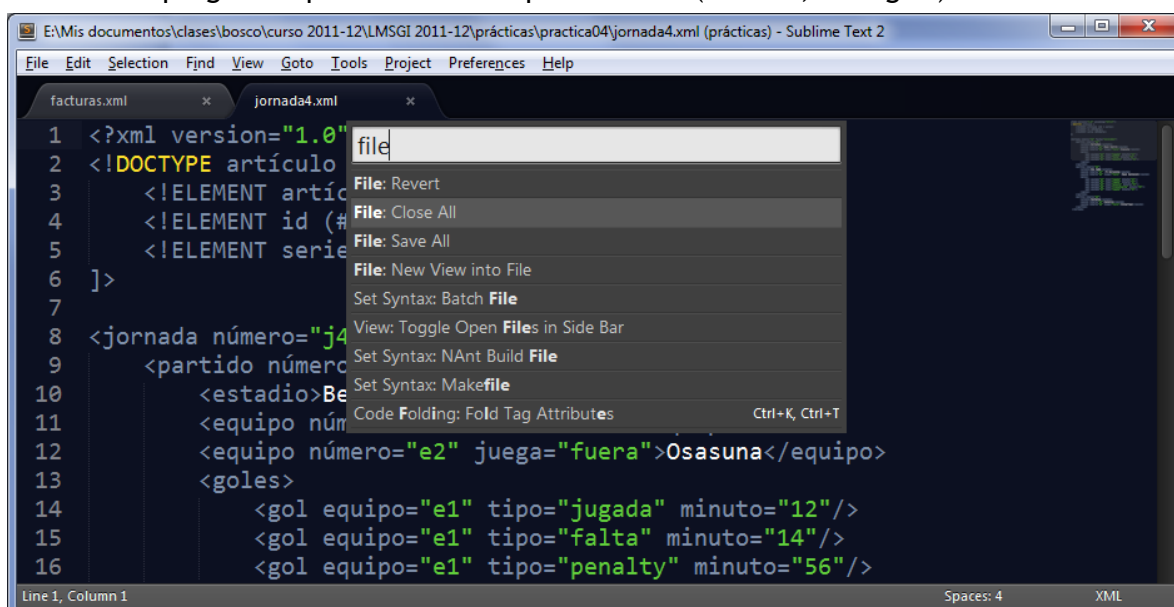


```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <persona>
3   <nombre>Jorge</nombre>
4   <apellidos>Sánchez</apellidos>
5   <telefono>666 666 666</telefono>
6   <telefono>999 999 996</telefono>
7 </persona>

theme loaded
reloading /E/Mis documentos/Sublime
Text/Data/Packages/User/Preferences.sublime-settings
found 1 files for base name Default.sublime-theme
theme loaded
  
```

- **View-Show Side Bar ó Ctrl+K, Ctrl+B** (es decir dejar pulsada la tecla **Ctrl** y pulsar primero **K** y luego **B**). Muestra la ventana lateral desde la que se pueden manejar los archivos abiertos o los pertenecientes a una carpeta (comentada en el apartado anterior).
- **Tools-Command Palette ó Ctrl+Mayús+P.** Ventana de comandos, permite buscar operaciones a realizar con el programa. Por ejemplo si pulsamos Ctrl+Mayús+P y luego escribimos **File**, nos aparecerán todas las operaciones del programa que contienen la palabra **File** (archivo, en inglés).



```

1 <?xml version="1.0"
2 <!DOCTYPE artículo
3 <ELEMENT artícu
4 <ELEMENT id (#
5 <ELEMENT serie
6 ]>
7
8 <jornada número="j4
9   <partido número
10    <estadio>Be
11    <equipo núm
12    <equipo número="e2" juega="fuera">Osasuna</equipo>
13    <goles>
14      <gol equipo="e1" tipo="jugada" minuto="12"/>
15      <gol equipo="e1" tipo="falta" minuto="14"/>
16      <gol equipo="e1" tipo="penalty" minuto="56"/>
  
```

- **Modo sin distracción.** Se consigue pulsando **Mayús+F11** o eligiendo **View-Select Distraction Free Mode**, con ello conseguimos quitar de la vista todas las paletas y menús para escribir sin estorbos. Pulsando de nuevo la tecla Mayús+F11 volvemos al modo normal
- **Pantalla completa.** Tecla **F11** o **View-Full Screen Mode**.

(3.4) edición de documentos

(3.4.1) multicursor

Se trata de una de las posibilidades más potentes del programa permite trabajar con varios cursores a la vez. Posibilidades

- **Ctrl+Clic de ratón.** Si pulsamos la tecla Ctrl a la vez que hacemos clic en diferentes sitios del documento, al escribir, escribiremos en varios sitios a la vez.

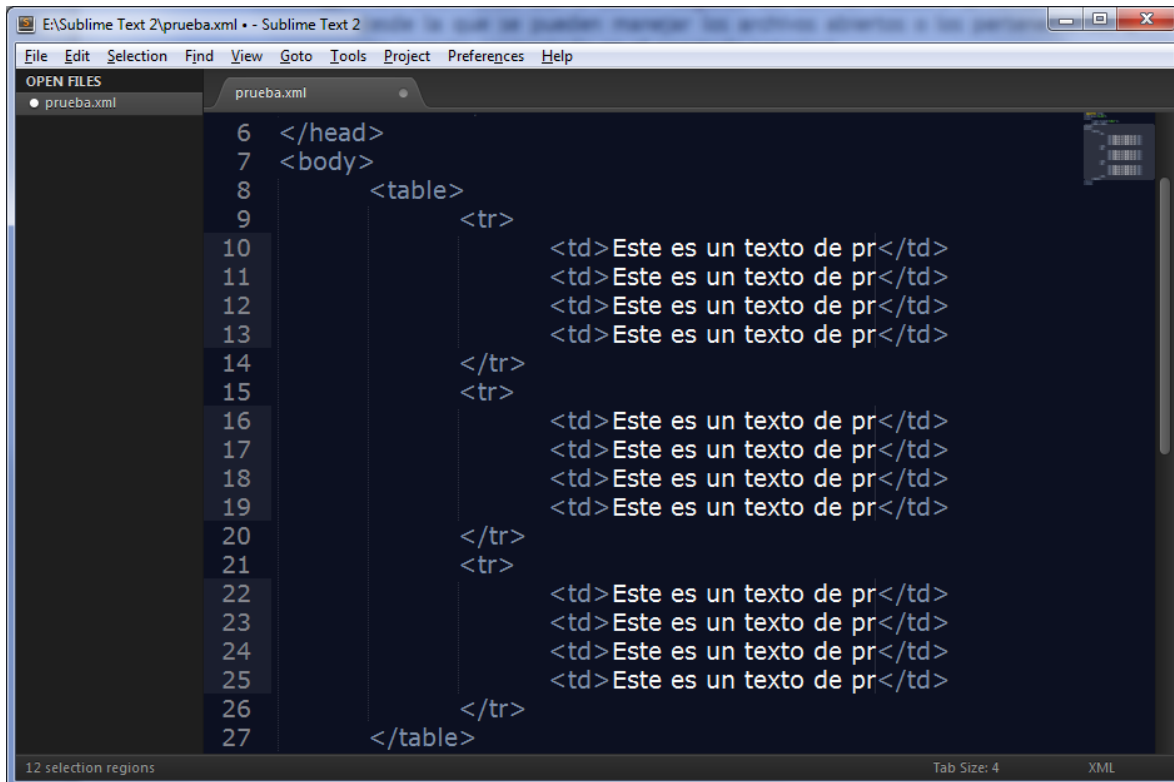


Ilustración 2, Texto escrito con el multicursor

- **Ctrl+Arrastrar con el ratón para seleccionar.** Permite seleccionar trozos de texto que no están juntos.

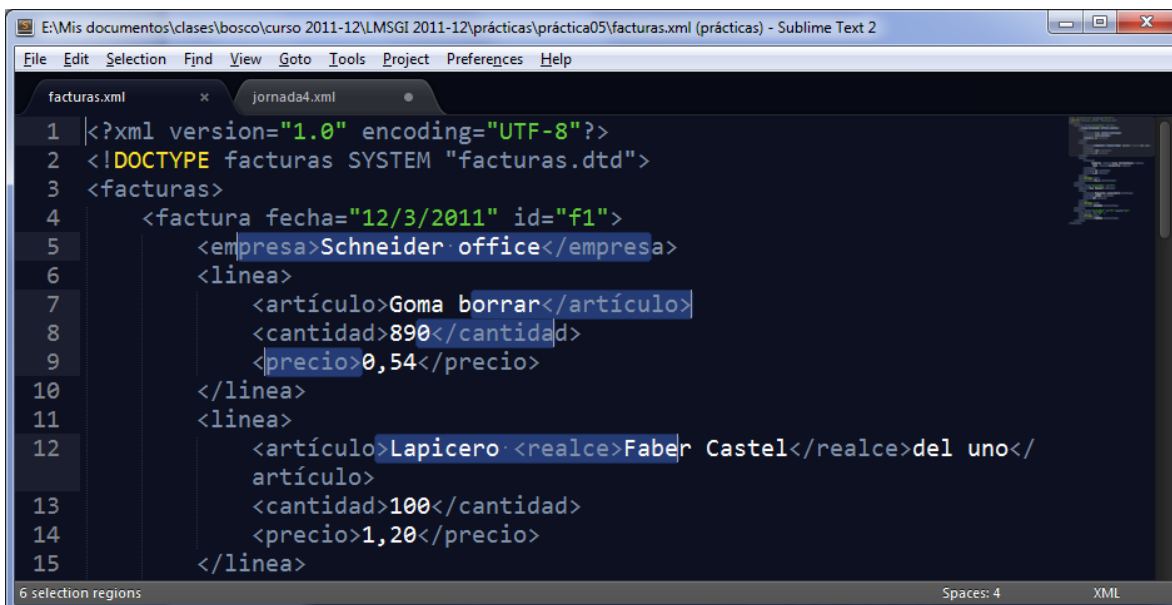


Ilustración 3, Selección usando tecla Ctrl a la vez que se arrastra

- **Ctrl+D.** Si seleccionamos una palabra en el documento, se marcan un poco todas las apariciones de esa palabra. Pulsando repetidamente Ctrl+D, podremos ir seleccionando sucesivamente todas las palabras como la nuestra. Con **Ctrl+U** las volver a deseleccionar una a una. Por ejemplo si nos ponemos delante de la palabra *hola* y vamos pulsando Ctrl+D repetidas veces, iremos seleccionando todos los *hola* del documento.

Mientras vamos seleccionando cada palabra con Ctrl+D, la combinación **Ctrl+K**, **Ctrl+D** nos permite saltar una de las selecciones. Es decir si tenemos cuatro veces la palabra *hola*, la Tecla Ctrl+D selecciona la primera, si queremos también la segunda volvemos a pulsar Ctrl+D, si la tercera nos la queremos saltar pulsamos Ctrl+K y luego Ctrl+D, si la cuarta sí la queremos seleccionar pulsamos Ctrl+D. Si ahora queremos anular esa última pulsamos Ctrl+U. Y así sucesivamente.

(3.4.2) teclas de selección de texto

- **Ctrl+L.** Selecciona la línea entera en la que tenemos el cursor.
- **Ctrl+Mayús+L.** Si tenemos seleccionadas varias líneas, coloca un cursor (en modo multicursor) en cada una de ellas.
- **Ctrl+Mayús+M.** Selecciona desde el símbolo de apertura de llave anterior (**{**), al símbolo de cierre de llave siguiente (**}**), respecto a la posición del cursor. Muy útil al escribir código en lenguajes de programación como **C**, **Java**, **C++**, **PHP**,...
- **Ctrl+Mayús+A.** Selecciona todas las líneas contenidas en la misma etiqueta que la actual. Útil en HTML o XML. Es decir selecciona todos los contenidos del elemento padre del actual.
- **Ctrl+Mayús+J.** Selecciona todas líneas cuya sangría (distancia al margen) esté dentro de la que nos influencia respecto a donde tenemos el cursor. Se usa muchísimo

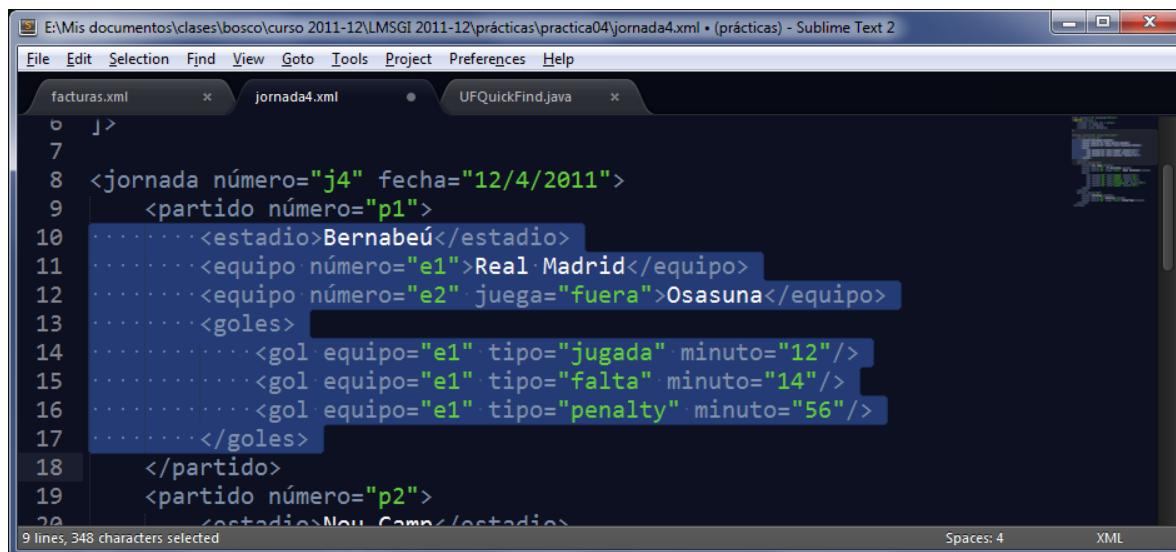


Ilustración 4, Selección con Ctrl+Mayús+J el cursor estaba en la primera línea con la etiqueta <equipo>

- **Ctrl+Mayús+Espacio.** Similar a Ctrl+Mayús+M, selecciona el elemento que contiene al cursor actual según las directrices del lenguaje en el que estamos escribiendo. Se le llama selección por alcance (*scope*)
- **Ctrl+Ç.** Colocar comentario, volviendo a pulsar se quita.
- **Ctrl+Mayús+Ç.** Si tenemos código seleccionado, pasa todo el código a comentario, pulsando otra vez se quita.

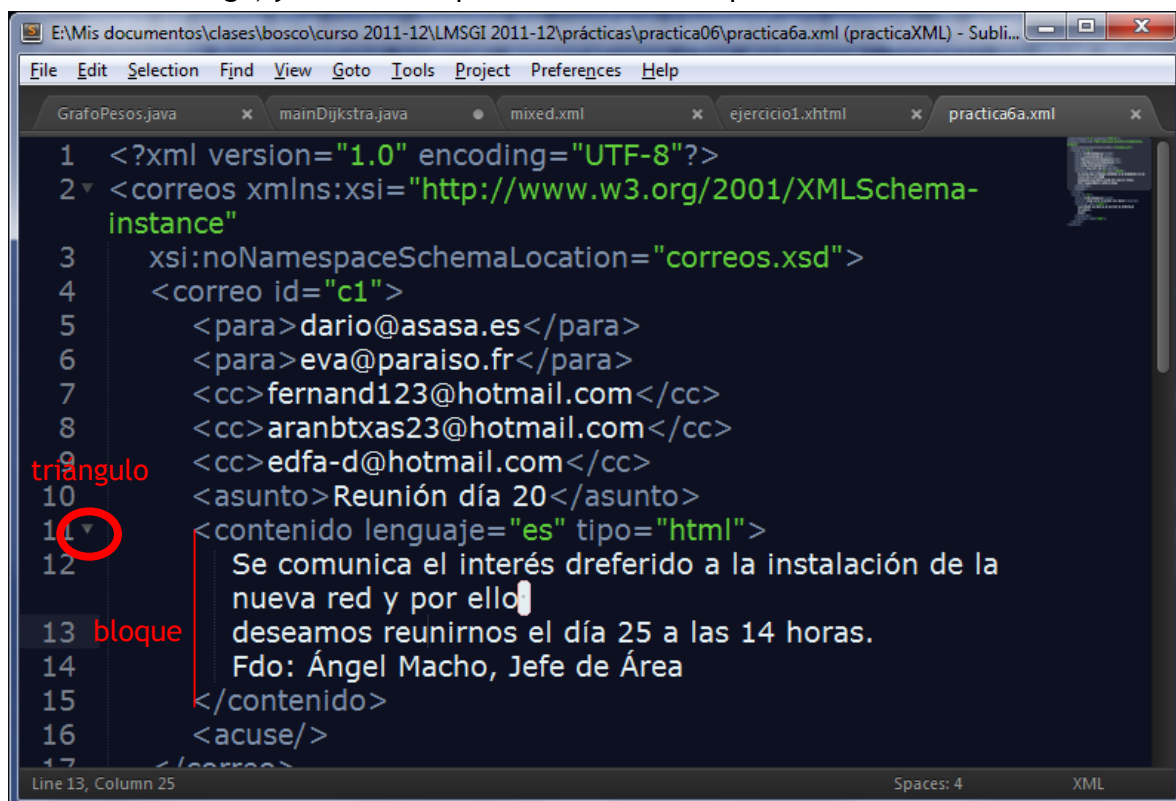
(3.4.3) teclas de edición de texto

- **Ctrl+Intro.** Inserta una línea bajo la actual.
- **Ctrl+Mayús+Intro.** Inserta una línea sobre la actual.
- **Ctrl+Flecha abajo.** Desplaza el documento hacia abajo una línea (scroll hacia abajo)
- **Ctrl+Flecha arriba.** Desplaza el documento hacia arriba una línea (scroll hacia arriba)
- **Ctrl+Mayús+Flecha arriba.** Sube la línea actual por encima de la anterior.
- **Ctrl+Mayús+Flecha abajo.** Baja la línea actual.
- **Ctrl+Supr.** Borra la palabra siguiente.
- **Ctrl+Retroceso.** Borra la palabra anterior.
- **Ctrl+Mayús+Supr ó Ctrl+K, Ctrl+K.** Borra hasta el final de la línea.
- **Ctrl+Mayús+Retroceso ó Ctrl+K, Ctrl+retroceso.** Borra hasta el principio de la línea.
- **Ctrl+Mayús+K.** Borrar la línea entera donde esté el cursor.
- **Ctrl+T.** Si tenemos varias cosas seleccionadas, las transpone, es decir las mueve de una a otra para intercambiar su posición.
- **Abrir corchete o llave en zona seleccionada.** Si tenemos algo seleccionado, al abrir llave o corchete (en lenguajes donde estos símbolos sirven para agrupar código), toda la selección quedará dentro de las llaves o corchetes.

- **Alt+Mayús+W.** Encierra el texto seleccionado dentro de un elemento HTML o XML. Tras pulsar las teclas luego se escribe el nombre y se pulsa Intro.
- **F9.** Ordena las líneas seleccionadas o todo el documento en orden alfabético.
- **Edit-Permute Lines.** Desde este menú podemos ordenar las líneas al revés de como estaban (opción **reverse**), quitar las líneas duplicadas (**unique**) u ordenar las líneas de forma aleatoria (**shuffle**)
- **Edit-Permute Selections.** Como la opción anterior pero referida al texto seleccionado.

(3.4.4) contraer código

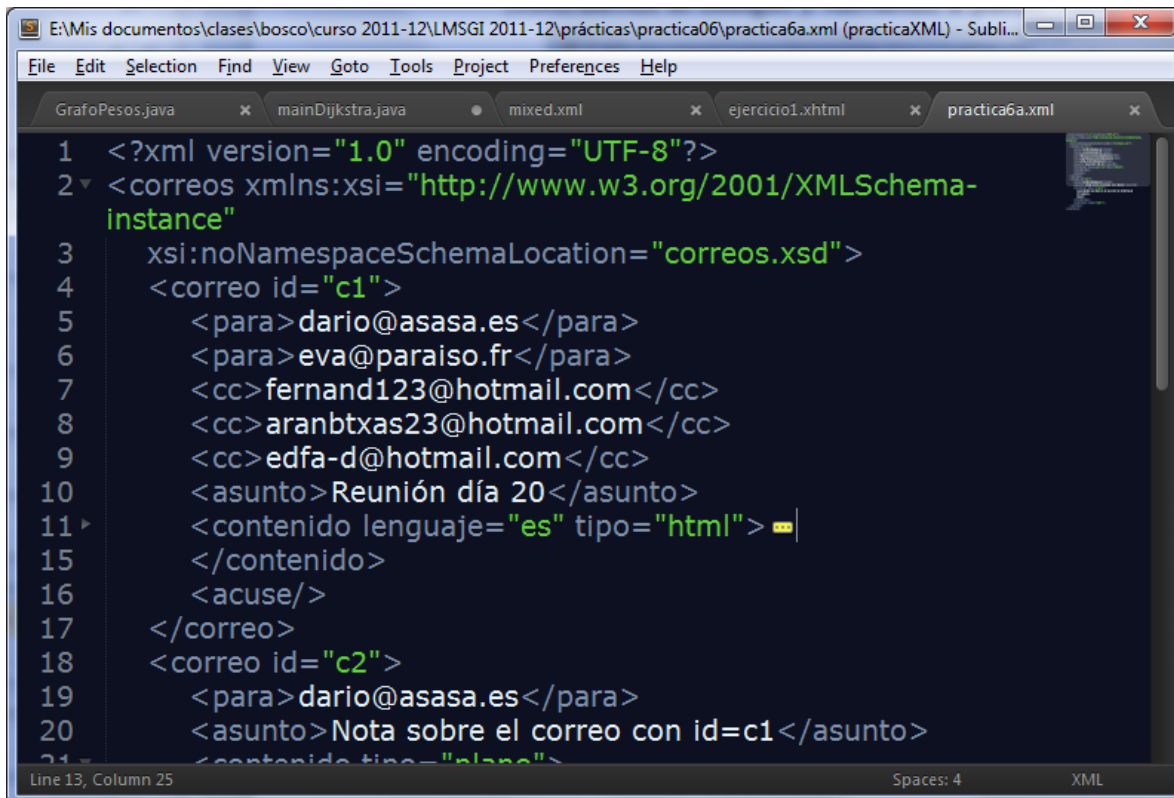
Estando dentro de un bloque de código (por ejemplo dentro de llaves en lenguajes como C, Java o PHP; o de etiquetas en XML o HTML), podemos señalar a la barra con los números de línea y aparece un triángulo; pulsándole ocultamos el código de ese bloque, (contraemos el código) y volviendo a pulsar volvemos a expandirle:



```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <correos xmlns:xsi="http://www.w3.org/2001/XMLSchema-
   instance"
3    xsi:noNamespaceSchemaLocation="correos.xsd">
4    <correo id="c1">
5      <para>dario@asasa.es</para>
6      <para>eva@paraiso.fr</para>
7      <cc>fernand123@hotmail.com</cc>
8      <cc>aranbtxas23@hotmail.com</cc>
9      <cc>edfa-d@hotmail.com</cc>
10     <asunto>Reunión día 20</asunto>
11     <contenido lenguaje="es" tipo="html">
12       Se comunica el interés drefenido a la instalación de la
13       nueva red y por ello
14       deseamos reunirnos el día 25 a las 14 horas.
15       Fdo: Ángel Macho, Jefe de Área
16     </contenido>
17   </correo>
18 </correos>
  
```

En la imagen anterior se ha indicado la posición del triángulo y la longitud del bloque, suponiendo que el cursor está dentro de la etiqueta **contenido**. Si se pulsa, el resultado sería:



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <correos xmlns:xsi="http://www.w3.org/2001/XMLSchema-
  instance"
3   xsi:noNamespaceSchemaLocation="correos.xsd">
4   <correo id="c1">
5     <para>dario@asasa.es</para>
6     <para>eva@paraiso.fr</para>
7     <cc>fernand123@hotmail.com</cc>
8     <cc>aranbtxas23@hotmail.com</cc>
9     <cc>edfa-d@hotmail.com</cc>
10    <asunto>Reunión día 20</asunto>
11    <contenido lenguaje="es" tipo="html">
12
13  </contenido>
14  <acuse/>
15 </correo>
16 <correo id="c2">
17   <para>dario@asasa.es</para>
18   <asunto>Nota sobre el correo con id=c1</asunto>
19   <contenido tipo="plain">
```

Se ha ocultado el texto que estaba dentro de la etiqueta **contenido**.

Otra forma de conseguir este efecto es con teclas y menús:

- **Ctrl+Mayús+'(apóstrofe)**. Si tenemos algo seleccionado, esta combinación oculta el código seleccionado para que no estorbe (no se borra, sólo se oculta. Es decir contrae el código seleccionado.
- **Ctrl+Mayús+í**. Nos muestra el código previamente contraído, es decir expande el código comprimido previamente.
- **Ctrl+k seguido de Ctrl+1**. Contrae todo el código referente al alcance superior respecto a la posición del cursor. Seguido de ctrl+2, contrae menos, Ctrl+3 aún menos y así sucesivamente.
- **Ctrl+K seguido de Ctrl+J**. Expande todo el código

(3.4.5) marcas

Las marcas permiten realizar operaciones de edición en base a la selección que indiquemos, son interesantes cuando queremos eliminar o seleccionar una gran cantidad de texto:

- **Ctrl+K, Ctrl+Espacio**. Pulsando una tras otra, ponemos marca a las líneas seleccionadas.
- **Ctrl+K, Ctrl+A**. Selecciona el texto desde la posición del cursor hasta la posición de la marca más próxima.
- **Ctrl+K, Ctrl+D**. Elimina el texto desde la posición actual hasta la posición de la marca más próxima.
- **Ctrl+K, Ctrl+G**. Quita todas las marcas

(3.4.6) teclas y acciones de edición de etiquetas (HTML, XML)

Disponibles en el menú **Edit-Tag**.

- **Alt+'.'** (punto). Cierra la última etiqueta abierta.
- **Alt+Mayús+W**. Teniendo texto seleccionado, encierra el texto dentro de la etiqueta cuyo nombre se indique.
- **Ctrl+Mayús+,** (coma). Coge la selección y la convierte en una etiqueta con su cierre.
- **Edit-Tag-Autoformat tag**. Recoloca las etiquetas para darles una forma visual más sencilla de leer
- **Edit-Tag-Remove Picked Tag**. Quita las etiquetas cuyo nombre se indica. Pueden ser varias separadas por espacios. Se pueden eliminar sólo las etiquetas que estén dentro de la selección de texto actual o todas las del documento.
- **Edit-Tag-Remove all tags**. Elimina todas las etiquetas (y no el texto normal) del documento o de la selección actual.
- **Edit-Tag-Remove...Attributes...** Funciona como los menús anteriores, pero referido a los atributos.

(3.4.7) uso de marcadores (bookmarks)

Los marcadores permiten señalar partes en el documento a las que luego podemos dirigirnos rápidamente. El manejo desde los menús está en **Goto-Bookmarks**
Posibilidades:

- **Ctrl+F2**. Quita y pone un marcador (los marcadores se dibujan con un triángi
- **F2**. Ir al siguiente marcador
- **Mayús+F2**. Ir al marcador anterior.
- **Ctrl+Mayús+F2**. Quita todos los marcadores
- **Alt+F2**. Selecciona todos los marcadores

(3.5) búsqueda

Es una de las operaciones fundamentales. Permite situarnos en el documento en la posición de un texto determinado o seleccionar ese texto. Todas las opciones fundamentales de búsqueda están en el menú **Find**.

(3.5.1) cuadro de búsqueda

La tecla **Ctrl+F** (menú **Find-Find**) permite estas posibilidades:

- (4) Escribir el texto a buscar
- (5) Pulsar **Find** (tecla **Intro**) para seleccionar la siguiente aparición en el documento del texto que busquemos.
- (6) Pulsar **Find Prev** (tecla **Mayús+Intro**) para seleccionar la aparición anterior en el documento del texto que buscamos.

- (7) Pulsar **Find All** (tecla **Alt+Intro**) para seleccionar todas las apariciones del texto en el documento (coloca una multiselección)

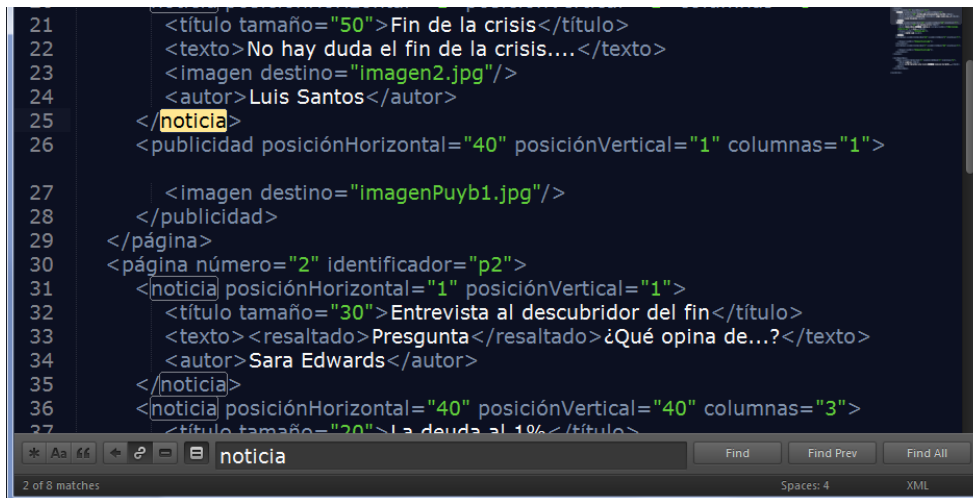


Ilustración 5, El cuadro de búsqueda de Sublime Text con la palabra **noticia** resaltada puesto que es la que hemos introducido en el cuadro. El resto de veces que aparece esa palabra está encuadrada.

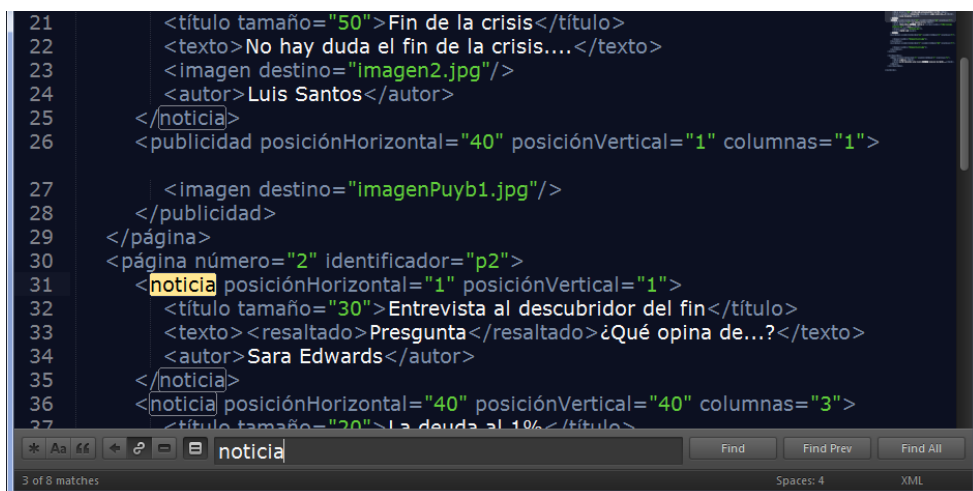


Ilustración 6, Situación tras pulsar **Find**. Se resalta la siguiente aparición de la palabra **"noticia"**

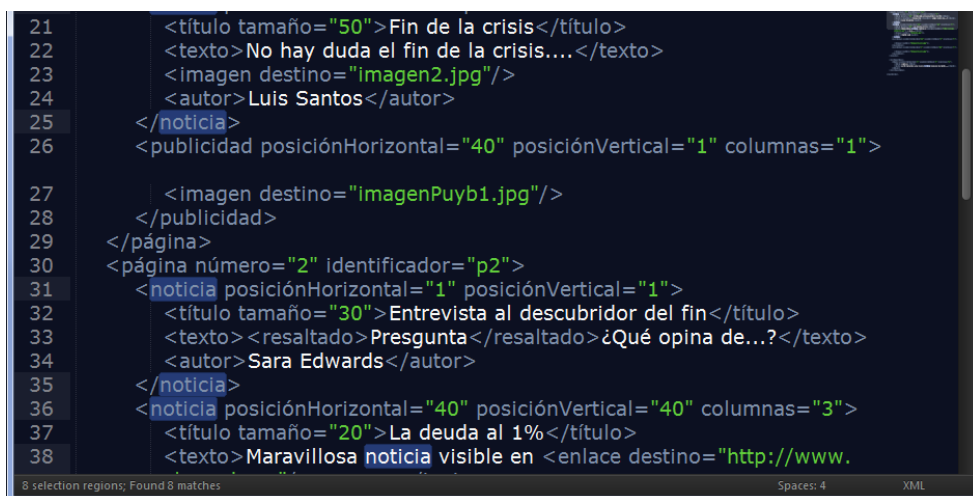


Ilustración 7, Tras pulsar **Find all**. Se resaltan todas las apariciones de la palabra **"noticia"**

la tecla **Ctrl+I** abre el mismo cuadro, pero en este caso la tecla **Intro** busca la siguiente aparición del texto a buscar, le selecciona y sale del panel de búsqueda.

(3.5.2) opciones avanzadas del cuadro de búsqueda

- Pulsando **Alt+R** o el botón **Regular Expression** (con un asterisco), podemos buscar en el documento el texto que encaje con una expresión regular que escribamos. Para saber más sobre expresiones regulares:
http://es.wikipedia.org/wiki/Expresi%C3%B3n_regular
- Pulsando **Alt+C** o el botón **case Sensitive** (A-a), sólo encuentra el texto si las mayúsculas y minúsculas están exactamente como las hemos escrito
- Pulsando **Alt+W** o **Exact Match** (") busca el texto sólo si es una palabra completa (por ejemplo si el texto buscado es la palabra **coche** no daría por buena la palabra **coches**).

(3.5.3) reemplazar

La tecla **Ctrl+H** (o el menú **Find-Replace**) permite buscar y reemplazar un texto. La tecla **Ctrl+Alt+Intro** cambia todas las apariciones del texto buscado por el texto que deseamos reemplazar.

(3.5.4) buscar en varios archivos

La tecla **Ctrl+Mayús+F** (o el menú **Find in Files**), saca una versión del cuadro de búsqueda que permite buscar y reemplazar texto en varios archivos (y no sólo en el que estamos). En principio busca en todos los archivos abiertos, la forma de hacerlo es:

- (4) Pulsar **Ctrl+Mayús+F**
- (5) Escribir el texto a buscar y pulsar **Intro**
- (6) Una nueva pestaña (llamada **Find results**) se abrirá para mostrar las veces que ha encontrado el texto y en qué archivos lo ha hecho.
- (7) Pulsando doble clic sobre el retazo deseado abriremos el archivo en la posición del texto buscado (también podemos ir de búsqueda en búsqueda pulsando **F4**)

(3.6) ir a cualquier parte

Sin duda una de las opciones más potentes para recorrer el texto en Sublime Text es usar el menú **Goto**, o mejor aún la tecla **Ctrl+P** (**Goto Anything**, *Ir a cualquier parte*).

(3.6.1) abrir documentos rápidamente

Con **Ctrl+P** aparece un panel con el que en principio podemos ir a otro documento de los que están abiertos o, más interesante, acceder rápidamente a uno de los archivos que se encuentren en la barra lateral (visible con **Ctrl+K, Ctrl+B**). Si hemos usado la opción **File-Open Folder** para abrir una carpeta entera, entonces el panel de ir a cualquier parte permitirá abrir fácilmente cualquier archivo de la carpeta, bien haciéndole clic en el panel o bien escribiendo su nombre (o parte del mismo).

(3.6.2) ir a números de línea

Si tras pulsar **Ctrl+P**, escribimos dos puntos (o elegimos **Goto-Goto line**), podremos escribir un número de línea y tras pulsar **Intro** iremos a dicho número.

(3.6.3) buscar texto

Si tras pulsar **Ctrl+P** escribimos el símbolo de la almohadilla (**#**), entonces aparece una lista con todas las palabras del documento y podremos rápidamente ir a ellas.

(3.6.4) buscar métodos

En lenguajes con posibilidad de definir funciones y procedimientos (como Java, C, C++, Python, ...) si tras el **Ctrl+P**, escribimos el símbolo de la arroba (**@**), aparece una lista de dichas funciones y podremos ir rápidamente a ellas.

(3.7) trabajar con proyectos

Un proyecto es un conjunto de archivos y carpetas con los que se desea trabajar como si fuera una unidad, de modo que se facilita la gestión conjunta de todos los archivos y carpetas del proyecto.

Las opciones de trabajo con los proyectos están en el menú **Project**. Normalmente la manera de empezar con el proyecto es abrir una carpeta preparada para tal fin mediante el menú **File-New Folder**. Se permitirá abrir una carpeta en una nueva ventana del programa. A partir de ahí podremos:

- Añadir más carpetas al proyecto, mediante **Project-Add new folders to project**. De esa manera el proyecto puede abarcar varias carpetas (con sus archivos)
- Quitar una carpeta del proyecto: en el panel lateral (**Ctrl+K, Ctrl+B**) pulsando un botón derecho sobre la carpeta y eligiendo la entrada en el menú **Remove folder from project**.
- Retirar todas las carpetas al proyecto mediante **Project-Remove all folders from Project**.

(3.7.1) guardar el proyecto

Cuando tengamos nuestro proyecto con las carpetas que deseamos manejar como una unidad entonces podemos guardar el proyecto mediante **Project-Save Project as**, tras lo cual se pedirá nombre y se guardará con el nombre elegido y la extensión de archivo **.sublime-project**.

Lo que graba el programa es un archivo de configuración en formato JSON que contiene al menos las rutas a los archivos. Para poder cambiar a mano dicho archivo se puede elegir la opción **Project-Edit project** y veremos su contenido que incluso podremos modificar (eso sí, sabiendo muy bien lo que hacemos).

(3.7.2) cerrar el proyecto

La opción **Project-close Project** cierra el proyecto y con ello, todos los archivos abiertos.

(3.7.3) abrir rápidamente un proyecto

la opción **Project-Switch to Project** o, mejor, la tecla **Ctrl+Alt+P** nos muestra una ventana con todos los proyectos que hemos creado. En ella podremos elegir el proyecto sobre el que queremos trabajar.

Al abrirle también nos dejará a la vista los archivos con los que estábamos trabajando en el proyecto.

También podremos abrir el archivo de proyecto desde el disco y tendrá el mismo efecto.

(3.7.4) modificar archivo de Proyecto

Con los proyectos, Sublime Text almacena un archivo con extensión **sublime-project** que contiene la información sobre el proyecto. Se trata de un archivo JSON que se puede modificar para añadir detalles al proyecto. Si abrimos un proyecto simple sólo suele aparecer esta información:

```
{
  "folders":
  [
    {
      "path": "/C/proyectos/xml"
    }
  ]
}
```

En este caso simplemente se indica la ruta a la única carpeta del proyecto (que sería la correspondiente a la ruta Windows **c:\proyectos\xml**).

El archivo de un proyecto abierto se puede modificar fácilmente desde los menús, eligiendo **Project-Edit Project**.

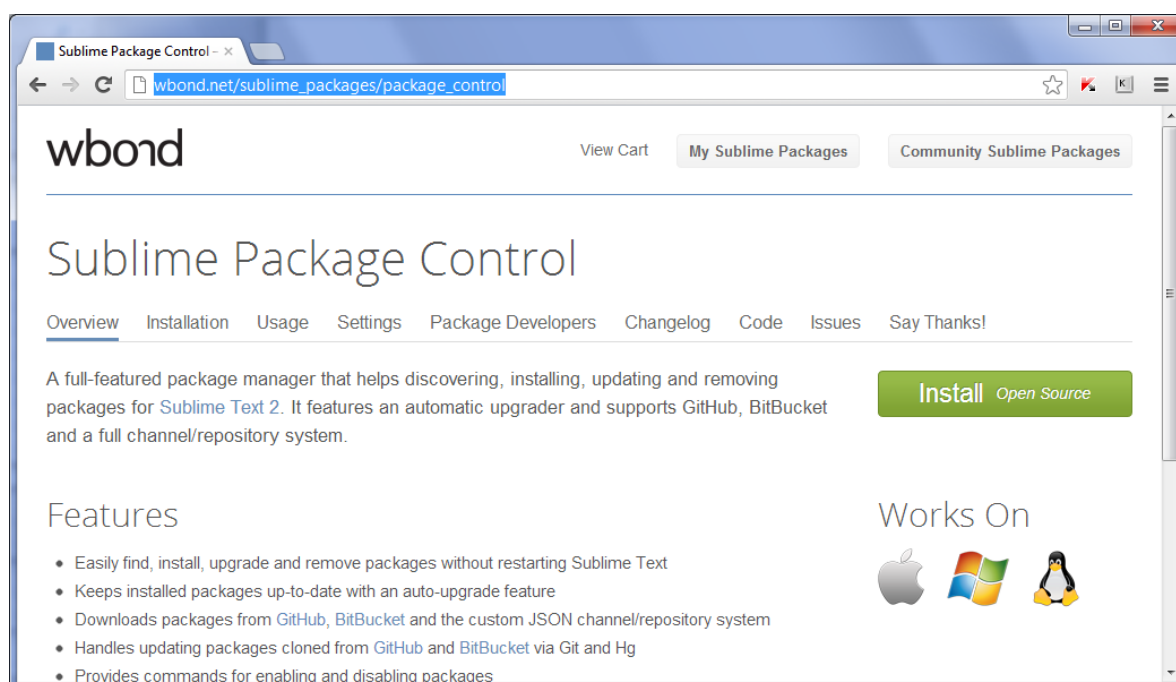
(4) instalación de paquetes

La potencia de Sublime Text se debe en gran parte a la posibilidad de añadir paquetes que potencian las posibilidades del editor. El número de paquetes es muy extenso, aquí se comentan algunos muy interesantes; se han tenido en cuenta los más interesantes para trabajar editando XML, HTML y PHP.

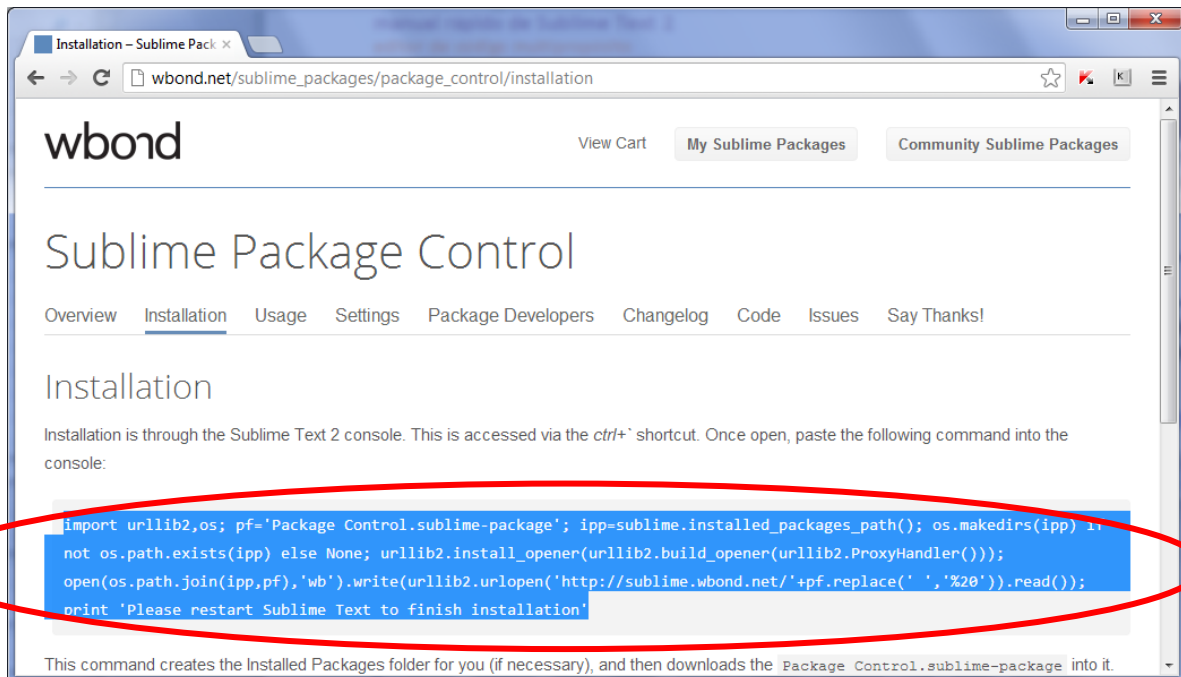
(4.1.1) Package Control

Lo primero que conviene hacer cuando se desea trabajar con paquetes es instalar un componente llamado Package Control que es un gestor fácil de paquetes. Es casi indispensable instalarlo. Lo malo es que algo difícil. La forma de hacerlo es:

- (1) Ir a Internet a la página http://wbond.net/sublime_packages/package_control

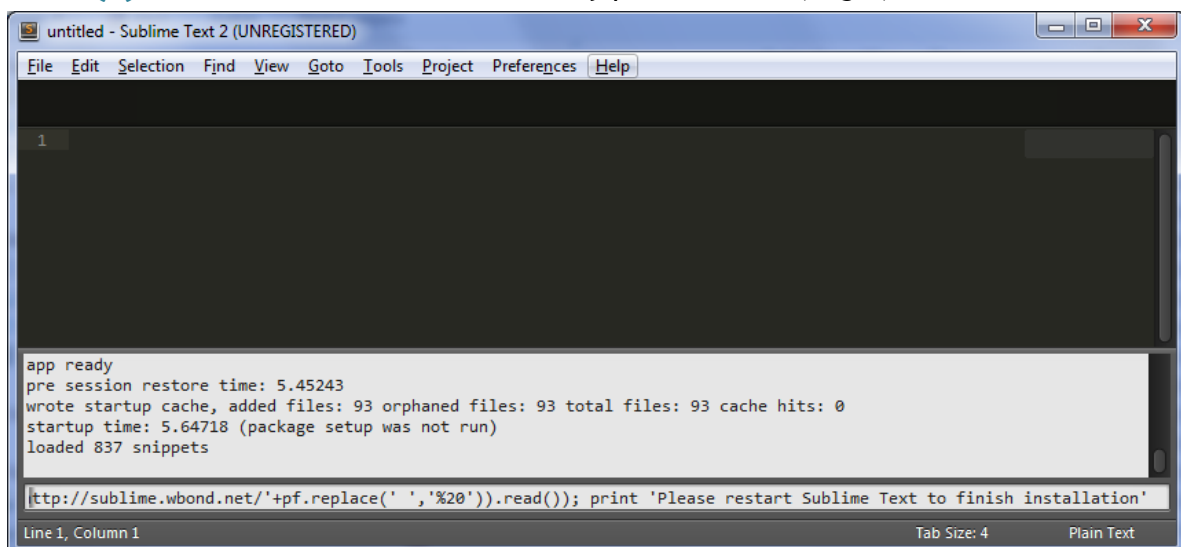


- (2) Hacer clic en el botón **Install Open Source**
- (3) En la página siguiente seleccionar el texto que está coloreado en gris (en código en lenguaje **Python**, entendible por el programa)
- (4) Pulsar las teclas **Ctrl+C** (copiar)



(5) Ir a Sublime Text y abrir la consola (**Ctrl+Ñ**) o **View-Show Console**.

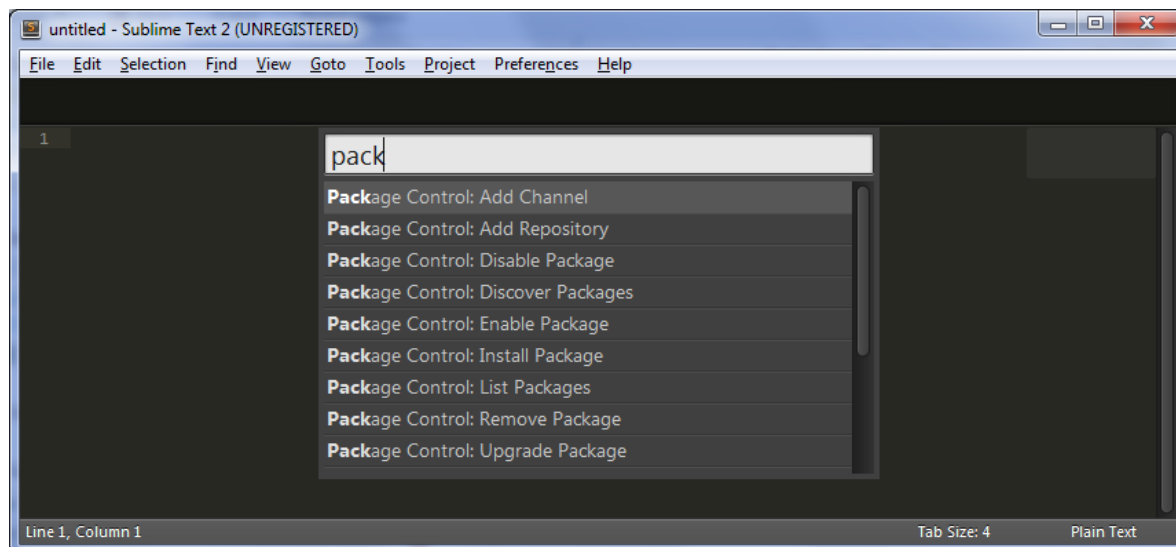
(6) Colocar el cursor en la consola y pulsar **Ctrl+V** (Pegar)



(7) Pulsar **Intro**. Se ejecutan las instrucciones, aparecerá el mensaje *Please restart Sublime Text to finish installation*

(8) Reiniciar el programa

(9) Si pulsamos **Ctrl+Mayus+P** (Paleta de Comandos) y escribimos *pack* se nos enseñarán (en inglés) todas las operaciones que podemos hacer con el Package Control.



Ahora podremos gestionar los paquetes que instalemos de forma más cómoda. En el panel de comandos (**Ctrl+Mayus+P**) podremos:

- Instalar un nuevo paquete con, **Package Control: Install Package**. Tras lo cual se nos mostrará otra ventana con todos los paquetes que se pueden instalar
- Mostrar la lista de paquetes instalados con **Package Control: List Packages**
- Eliminar un paquete instalado con **Package Control: Remove Package**

(4.1.2) instalar paquetes

Para instalar paquetes una vez instalado el magnífico **Package Control**, se haría lo siguiente:

- (1) Abrir el panel de comandos (**Ctrl+Mayus+P**)
- (2) Elegir **Package Control-Install Package** (si escribimos **pack** al aparecer la paleta de comandos, será más fácil encontrar la opción de instalación)
- (3) De la lista de paquetes que aparece (y que el programa obtiene de Internet), elegir el paquete deseado.

(4.1.3) paquetes interesantes (especialmente para PHP, CSS y XML/HTML)

Tag

Pensado para escribir en HTML/XML, facilita el manejo de etiquetas. Una vez instalado (desde el **Package Control**, eligiendo el paquete **tag**). Desde la paleta de comandos (**Ctrl+Mayus+P**) escribiendo **Tag** encontraremos todas las posibilidades de manejo con etiquetas. Algunas tienen teclas asociadas, la mejor quizá **Ctrl+Alt+T** que coloca correctamente las sangrías del documento HTML o XML.

ZenCoding

Uno de los más potentes paquetes. La única pega es que quita funcionalidades a algunos de los otros que hayamos instalado, pero compensa con creces este problema. Una vez instalado desde el **Package Control**, en cada archivo HTML y XML que usemos podremos

escribir abreviaturas que nos facilitan la escritura en estos lenguajes. Todas se basan en escribir un texto y pulsar el tabulador.

De entrada disponemos de estas abreviaturas (en inglés **snippets**):

- **html:5**. Escribiendo exactamente `html:5` (incluidos los dos puntos) y pulsando el tabulador, se nos escribirá la plantilla de un documento HTML 5 completo.
- **html:xt**. Plantilla para XHTML 1.0 transicional.
- **html:xs** Plantilla para XHTML 1.0 estricto
- **html:xxs** Plantilla para XHTML 1.1 estricto
- **html:4s** Plantilla para HTML 4 estricto
- **html:4t** Plantilla para HTML 4 Transicional

Hay muchas más, todas se pueden ver en el archivo **zen_settings.py** (archivo que hay que examinar con cuidado) que se encuentra en el directorio **ZenCoding** dentro del paquete **ZenCoding** en la carpeta **packages** en el directorio en el que se haya instalado el Sublime Text.

Sin duda lo más espectacular de este paquete es la posibilidad de escribir abreviaturas para escribir código XML o HTML muy rápido. Ejemplo (en un documento XML):

- Escribiendo **documento** y luego pulsando tabulador, Zen Coding lo cambia por:

```
<documento>

</documento>
```

- Escribiendo **documento+notas**, se cambia por (tras pulsar el tabulador):

```
<documento>

</documento>
<notas>

</notas>
```

- Escribiendo **documento[título]** se cambia por:

```
<documento titulo="">

</documento>
```

- Escribiendo **documento[título="docu1"]** se cambia por:

```
<documento titulo="docu1">

</documento>
```

- Escribiendo **documento>autor**

```
<documento>

<autor>
```

```
</autor>  
</documento>
```

■ Escribiendo *documento>autor>nombre*

```
<documento>  
  <autor>  
    <nombre>  
  
    </nombre>  
  </autor>  
</documento>
```

■ Escribiendo *documento>autor>nombre+apellidos*

```
<documento>  
  <autor>  
    <nombre>  
  
    </nombre>  
    <apellidos>  
  
    </apellidos>  
  </autor>  
</documento>
```

■ Escribiendo *documento>autor*3*

```
<documento>  
  <autor>  
  
  </autor>  
  <autor>  
  
  </autor>  
  <autor>  
  
  </autor>  
</documento>
```

■ Escribiendo *documento>titulo+texto>tipo+linea*5*

```
<documento>  
  <titulo>  
  
  </titulo>  
  <texto>  
    <tipo>
```

```
</tipo>
<linea>

</linea>
<linea>

</linea>
<linea>

</linea>
<linea>

</linea>
<linea>

</linea>
</texto>
</documento>
```

Y así podemos seguir haciendo combinaciones en las que es posible utilizar paréntesis para aclarar bien la expresión.

Clipboard History

Se trata de un componente que permite usar un portapapeles de múltiples elementos. Es uno de los paquetes más útiles. Cada vez que copiamos texto del documento al portapapeles (con la tecla **Ctrl+C** por ejemplo), este componente le almacena, de modo que todo lo que hemos copiado se queda en memoria.

Sacando la paleta de comandos (**Ctrl+Mayús+P**) y luego eligiendo **Clipboard History**, nos aparece una última paleta en la que dispondremos de todo lo copia para pegarlo de nuevo.

La paleta del **Clipboard History** sale más rápido mediante la tecla **Ctrl+Alt+V**, así accedemos más rápido a los textos copiados.

BracketHighlighter

Se trata de un paquete que permite colorear de forma más resaltada los elementos que agrupan código (por ejemplo las etiquetas HTML o las llaves de los lenguajes como Java, C, PHP,...). Gracias a él se hace más difícil olvidarse de cerrar una etiqueta, una llave, un corchete, etc.

Lo malo es que el coloreado se hace demasiado fuerte por lo que conviene a veces modificar los colores que trae predeterminados en sus archivos de configuración.

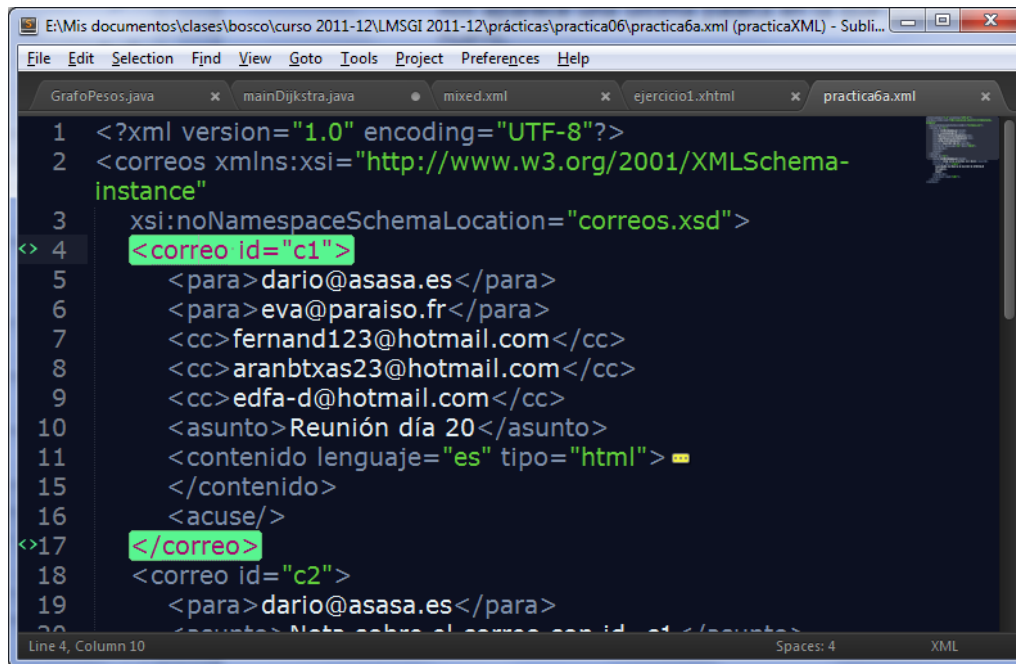


Ilustración 8, BracketHighLighter iluminando la etiqueta sobre la que nos hemos posicionado para que se vea claramente donde está la apertura y el cierre de la misma

Trailing Spaces

Se trata de un paquete que resalta los llamados trailing spaces, que son los espacios innecesarios en el código al final de las líneas y que incomodan en el mismo (además de ocupar espacio innecesario).

Una vez instalado, desde el package control, los espacios innecesarios se resaltan cada vez que los detecta el paquete. Para borrarlos basta con elegir **Ctrl+Mayus+P** (paleta de comandos) y buscar **Trailing Spaces>Delete** (escribiendo por ejemplo **delete** en la paleta para filtrar el texto que nos muestra).

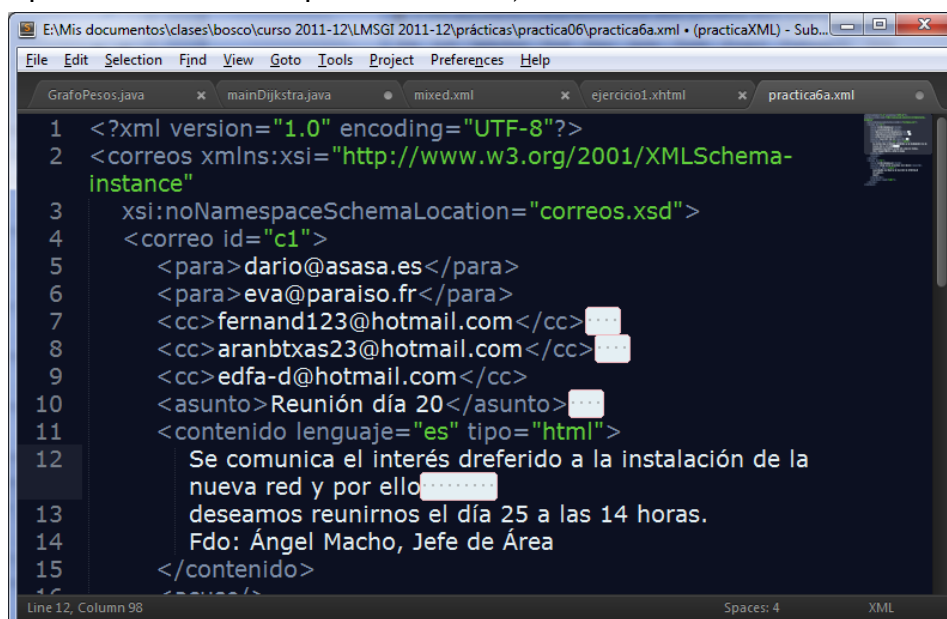


Ilustración 9, Espacios innecesarios al final resaltados en blanco por el paquete Trailing Spaces

DocBlockr

Es un paquete muy interesante, para PHP y JavaScript que permite documentar el código al estilo de la documentación del lenguaje Java (conocida como **JavaDoc**). Una vez instalado funciona muy fácil, nos ponemos por encima de una función o una clase de PHP o JavaScript y escribimos **/**** en seguida nos resalta el texto al que se referirá el comentario. Al pulsar la tecla Intro, nos facilitará la escritura del comentario y de ese modo la documentación (tan importante) del mismo.

Sublime Linter

Un **Linter** es un programa que nos corrige los errores a medida que escribimos. de esta forma, aún sin validar nuestro programa, desde el propio Sublime Text podremos ir viendo si nuestro código es limpio o no. **Sublime Linter** es el mejor Linter para Sublime Text, pero depende de que tengamos instalado en nuestro ordenador el programa apropiado para verificar el código. Es decir necesitamos un programa estándar de validación de código para el lenguaje que deseamos validar.

En concreto es capaz de validar código **PHP, Ruby, Python, HTML, HAML, XML, CSS, CoffeeScript, Java, JavaScript** y **Perl**. Pero, como se ha comentado, buscará el validador estándar de esos lenguajes; además dicho validador debe de ser accesible desde la línea de comandos del sistema y eso significa añadir a la variable **Path** del sistema la ruta a dicho validador (véase **Apéndice: Modificación del Path del sistema**).

Ejemplos:

- Para validar **PHP** SublimeLinter llama al programa **php** este programa está en la carpeta en la que hayamos instalado PHP y por lo tanto deberá estar en el path de Windows.
- Para XML debe estar disponible el programa **xmllint**, pero el paquete Tag al menos nos mostrará los errores que impidan que el documento XML no esté bien formado.
- Para HTML utiliza **tidy** descargable para Windows en formato ZIP en <http://tidybatchfiles.info/>
- etc. (toda esta información está en la ayuda del paquete, archivo **README**).

Ctag

Exhuberant CTags es un programa que permite realizar un índice de las funciones y código de un proyecto para facilitarnos la navegación por el mismo. Está pensado para todo tipo de lenguajes; pero no para lenguajes de etiquetas como HTML o XML.

Sublime Text dispone de dos paquetes para trabajar con **Ctags**, uno es de propósito general (para todo tipo de lenguajes como C, Java, Ruby, Python,...) y hay otro específico para PHP (que requiere primero haber instalado el anterior).

Los pasos para utilizar este interesante paquete son:

- (1) Descargar **Exhuberant CTags** desde la página <http://ctags.sourceforge.net/> en la parte final de la página aparece el archivo ZIP para descargar en el caso de la versión Windows (es posible instalarle en sistemas Linux mediante código fuente o paquete **rpm**).
- (2) Una vez descargado, descomprimirle en el sitio que deseemos y la carpeta del programa añadirle al **Path**. Los pasos son los indicados en el apéndice final: **Modificación del Path de Windows**. De otro modo el paquete no funciona al no encontrar el programa que permite su funcionamiento

- (3) Ya desde Sublime Text, instalar el paquete **CTag** usando **Package Control**.
- (4) Instalar después el paquete **CTag para PHP** especialmente dedicado a este lenguaje.
- (5) Abrir un proyecto y desde la ventana lateral del mismo (**View-Side Bar**) pulsar el botón derecho del ratón sobre la carpeta encima de todas las proyecto y elegir **CTags:Rebuild tags**. Esa opción crea un índice con todos los elementos que se han definido en el código.
- (6) Cada vez que estemos en un documento del proyecto, podemos pulsar el botón derecho del ratón sobre un elemento (variable, función, clase,...) del código y podremos elegir **Navigate to Definition**, Sublime Text nos llevará hasta la posición en la que se define ese elemento, no importa si está en otro archivo: Sublime Text le abrirá.

Esta es la potencia de este paquete que permite navegar de forma muy cómoda por el código. Si pulsamos **Ctrl+T, Ctrl+T** (es decir dejar pulsado **control** y pulsar la letra **T** dos veces), iremos aún más rápido a la definición.

Prefixr

Es un paquete especialmente pensado para CSS3. De modo que si se usa una sintaxis CSS3 correspondiente a un navegador (como Explorer), el paquete añade automáticamente en el archivo CSS las opciones equivalentes y correspondientes a los otros navegadores (como Firefox, Safari,...).

Todos los desarrolladores en CSS3 saben del tiempo que ahorra esta opción. Para funcionar con ella se escribe el CSS y se pulsa la tecla **Ctrl+Alt+X** y entonces busca el CSS que tiene que multiplicar para cada navegador.

Goto Documentation

Se trata de un paquete imprescindible para los programadores de PHP. Permite consultar la documentación oficial referida al lenguaje que estamos escribiendo. Sólo funciona (por ahora) para PHP, JavaScript/CoffeeScript y Ruby on Rails.

Una vez instalada (usando **Package Control: Install Package** y luego eligiendo **Goto Documentation**, para usarla basta con seleccionar algo de nuestro código sobre lo que deseamos documentarnos y entonces sacar la paleta de comandos (**Ctrl+Mayús+P**) y elegir **Goto Documentation**, entonces a través de nuestro navegador se nos mostrará la documentación oficial existente sobre el texto seleccionado.

Para facilitar aún más el uso de este interesante paquete, lo lógico es asignarle una tecla. Para lo cual habría que cambiar el archivo de teclado (**Settings-Keybindings User**), para añadir por ejemplo la línea:

```
{ "keys": ["shift+f1"], "command": "goto_documentation" }
```

De esa forma al pulsar **Mayús+F1** sobre el texto seleccionado, lanzaremos la ayuda sobre ese texto.

(5) personalizar Sublime Text

Otra de las potencias del programa es la posibilidad de intervenir en cada aspecto de su funcionamiento. Es lo que se conoce como personalizar (*custom*) el entorno. Los ajustes se guardan en diferentes archivos que se almacenan en la carpeta **user** dentro de los paquetes de Sublime Text.

(5.1) ajustes y teclas

(5.1.1) personalizar ajustes

Se puede personalizar todo el funcionamiento de Sublime Text. Esto es posible manejando el menú **Preferences**.

En él aparece un apartado llamado **Settings-Default** (ajustes por defecto). Si le marcamos, se abrirá el archivo de ajustes del programa. No conviene modificar nada aquí, pero en este archivo (con formato **JSON**) se observan todas las configuraciones del programa que funcionan en la forma:

```
"clave":"valor"
```

Aunque hay claves complicadas que requieren más valores (para entender bien los símbolos habría que conocer bien JSON, pero por intuición se puede entender bastante bien).

Realmente el archivo de ajustes que contiene nuestras preferencias está en el mismo apartado pero eligiendo **Settings-User**. Al principio está vacío pero podemos añadir las claves y valores que queramos (para lo cual conviene haberlas examinado en el **Settings-Default**), nuestras claves tienen preferencia sobre las del archivo por defecto, por eso basta con tocar el archivo **Settings-Users**. Si algo no funcionara bien, bastaría con borrar el contenido del archivo para volver a la configuración inicial. Ejemplo de archivo **Settings-Users**:

```
{
  "color_scheme": "Packages/Color Scheme - Default/Blackboard.tmTheme",
  "font_size": 14,
  "font_face": "verdana",
  "theme": "Soda Dark.sublime-theme",
  "soda_classic_tabs":true
}
```

Se marca en él que usamos el esquema de colores **Blackboard**, que el tamaño de letra será 14 y la letra Verdana, que usamos el tema Soda Dark y que además usamos en él la forma clásica de pestaña.

(5.1.2) personalizar teclas

El funcionamiento es semejante, el archivo **Key Bindings-Default** contiene la configuración de teclas por defecto del programa. Es otro archivo que tiene decenas de entradas en esta forma:

```
"keys":["tecla"],"command":"nombre del comando"
```

Resulta difícil saber el formato y como antes el archivo por defecto no conviene modificarlo. Por ello lo que se suele hacer es copiar la entrada de tecla que queremos modificar y Abrir el archivo de teclas de usuario: **Preferences-Key Bindings-User**. Ejemplo de archivo con teclas de usuario:

```
[  
  { "keys": ["ctrl+alt+t"], "command": "tag_indent_document" },  
  { "keys": ["ctrl+alt+c"], "command": "toggle_comment", "args": { "block": false } },  
  { "keys": ["ctrl+shift+alt+c"], "command": "toggle_comment", "args": { "block": true } }  
]
```

En este caso la combinación **Ctrl+Alt+T** reformatea la configuración de sangrías del código, mientras que, por ejemplo, **Ctrl+Alt+C** sirve para colocar un comentario usando el texto de la posición actual.

(5.1.3) personalizar ajustes en paquetes

También podemos realizar ajustes sobre los paquetes instalados. En el mismo menú de **Preferences** podemos elegir **Package Settings** y ahí aparecerán todos los paquetes instalados a los que le podemos realizar los mismos ajustes que al programa en general.

La idea es la misma observar la configuración en uso en el apartado **Default**, y modificar el **User** para escribir los cambios que deseemos.

(5.2) personalizar colores y temas

(5.2.1) personalizar temas

El menú de preferencias (**Preferences**) en el apartado **Color-Scheme** dispone de los temas predefinidos que utiliza el programa, si simplemente elegimos un esquema cambian los colores del código.

Pero es posible personalizar esos esquemas a fin de crear uno propio. Realmente los esquemas se graban en archivos XML con extensión **tmTheme**. Para crear un tema personal, la opción más cómoda es:

- (1) Abrir el tema deseado original usando la ruta:

\Data\Packages\Color Scheme - Default

a partir de la carpeta en la que esté instalado el programa. En esa carpeta se encuentran todos los temas, abrir en Sublime Text el archivo **tmTheme** que tenga los ajustes del tema que más se parezca a lo que deseamos.

- (2) Guardar el archivo en la carpeta **User** dentro de **Packages** con el nombre que deseamos para nuestro tema y respetando la extensión **tmTheme**.

- (3) Modificar nuestro archivo cambiando colores y lo que deseemos (eso sí entendiendo un poco el código del archivo)
- (4) Guardar de nuevo
- (5) El tema personal aparece en los menús del programa en **Preferences-Color Schema-User** podremos elegirle cuando queramos.

(5.2.2) personalizar colores de otros paquetes

En otros paquetes puede haber temas usados de la misma forma. Podemos también usar nuestro tema para colorear aspectos de otros paquetes para lo cual hay que saber como funcionan. Eso supera el objetivo de esta guía, pero buscando en los sitios adecuados no es tan difícil hacerlo.

(5.3) macros

Como en otros programas informáticos, las macros son una serie de operaciones que se graban para poder ser ejecutadas de golpe y así automatizar su uso.

En el menú **Tools** podemos echar un vistazo a las macros en el menú llamado así, **Macros**.

Si queremos grabar una macro los pasos son:

- (1) Elegir **Tools-Record Macro** (o **Ctrl+Q**)
- (2) hacer las operaciones que queremos asociar a la macro (por ejemplo subir la letra, alinear el código,...)
- (3) Una vez realizadas, volver a pulsar **Ctrl+Q** o elegir **Tools-Stop recording macro**
- (4) Guardar la macro con **Tools-Save Macro**. Grabará un fichero en la carpeta de usuario con el nombre que indiquemos y la extensión (que debemos respetar) **.sublime-macro**.

Después podremos editar la macro, que también es un archivo de tipo JSON. Podremos invocar a la macro las veces que haga falta con **Tools-Macro-nombre de la macro**

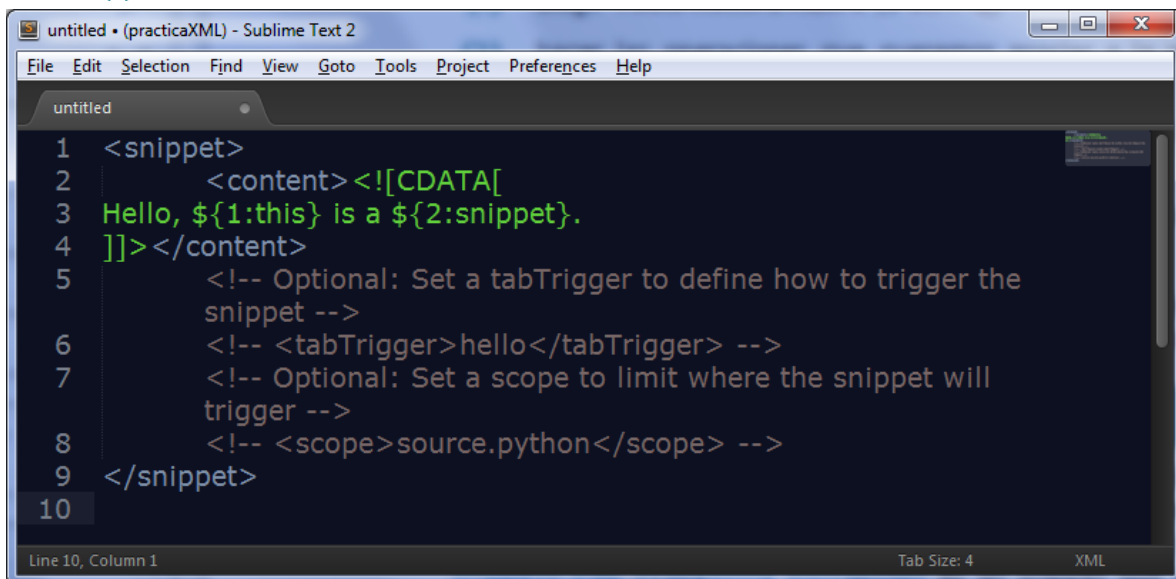
(5.4) snippets

Los snippets son trozos de código que se invocan para facilitar la tarea. Por ejemplo si estamos en un archivo XML y abrimos la paleta de comandos (**Ctrl+Mayús+P**) y luego escribimos Snippet, aparecen unos cuantos snippets ya definidos, por ejemplo **Snippets:XML Processing Instruction** es un snippet que permite escribir de golpe el inicio de un documento XML: **<?xml versión="1.0" encoding="UTF-8"?>**.

Si deseamos crear nuestros snippets, el proceso parece complejo pero si le cogemos el aire es muy fácil. Hay que tener en cuenta que un snippet es un trozo de código que vamos a utilizar de manera más o menos habitual, lo fundamental es que el código sea correcto. Además nos conviene pensar bien en el snippet para que nos facilite al máximo el trabajo

Para crear un snippet hay que

(1) Pulsar en **Tools-New Snippet**



The screenshot shows the Sublime Text 2 interface with a file named 'untitled' open. The menu bar includes File, Edit, Selection, Find, View, Goto, Tools, Project, Preferences, and Help. The text area contains an XML snippet template with the following content:

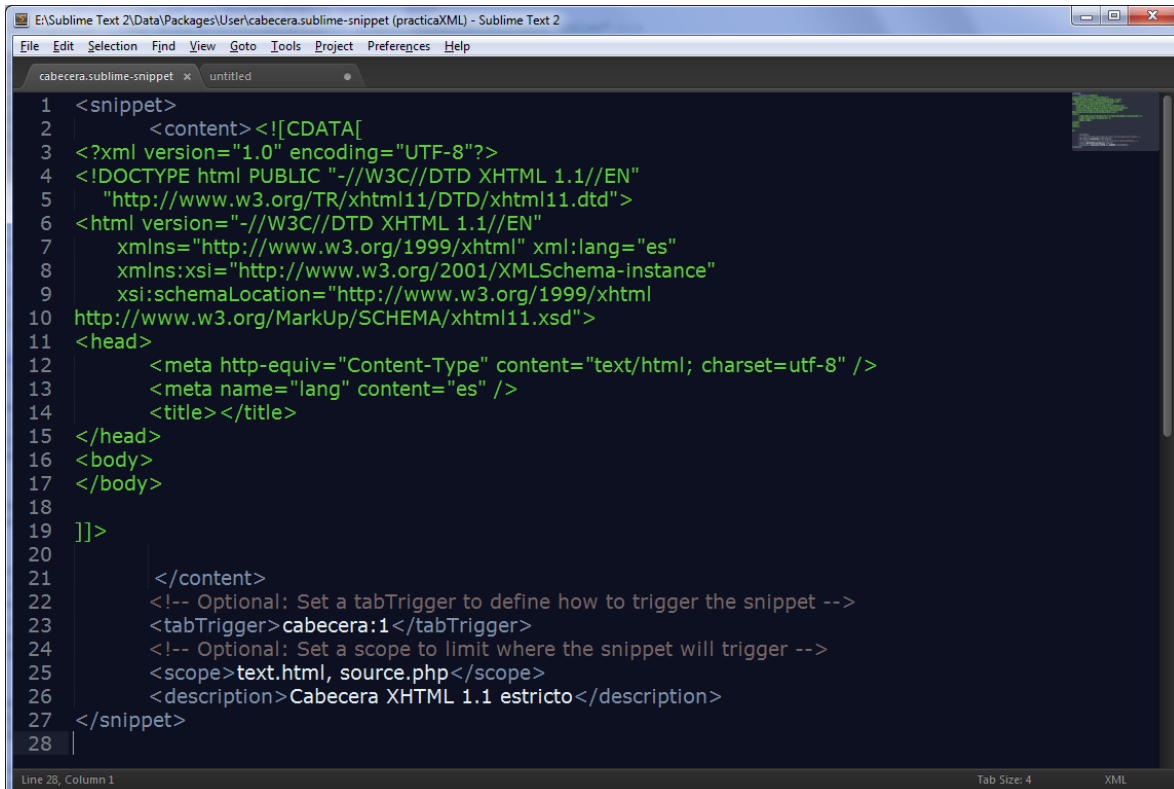
```
1 <snippet>
2   <content><![CDATA[
3   Hello, ${1:this} is a ${2:snippet}.
4   ]]></content>
5   <!-- Optional: Set a tabTrigger to define how to trigger the
6       snippet -->
7   <!-- <tabTrigger>hello</tabTrigger> -->
8   <!-- Optional: Set a scope to limit where the snippet will
9       trigger -->
10  <!-- <scope>source.python</scope> -->
11 </snippet>
```

The status bar at the bottom indicates 'Line 10, Column 1', 'Tab Size: 4', and 'XML'.

(2) Aparece la ventana anterior, en esa ventana debemos:

- Dentro del **content** en el apartado **CDATA** colocar el código deseado.
- Quitar los comentarios (**Ctrl+C**) del apartado **tabTrigger** y escribir un texto corto que nos permita invocar al snippet de manera fácil. Ese texto funcionará de modo que si lo escribimos y pulsamos la tecla tabulador, le sustituirá por el código del snippet.
- En el apartado **scope** podemos quitar los comentarios e indicar en qué tipo de documentos funciona el snippet.
- Podemos añadir un apartado **description** para poner un texto que describa al snippet

Ejemplo:

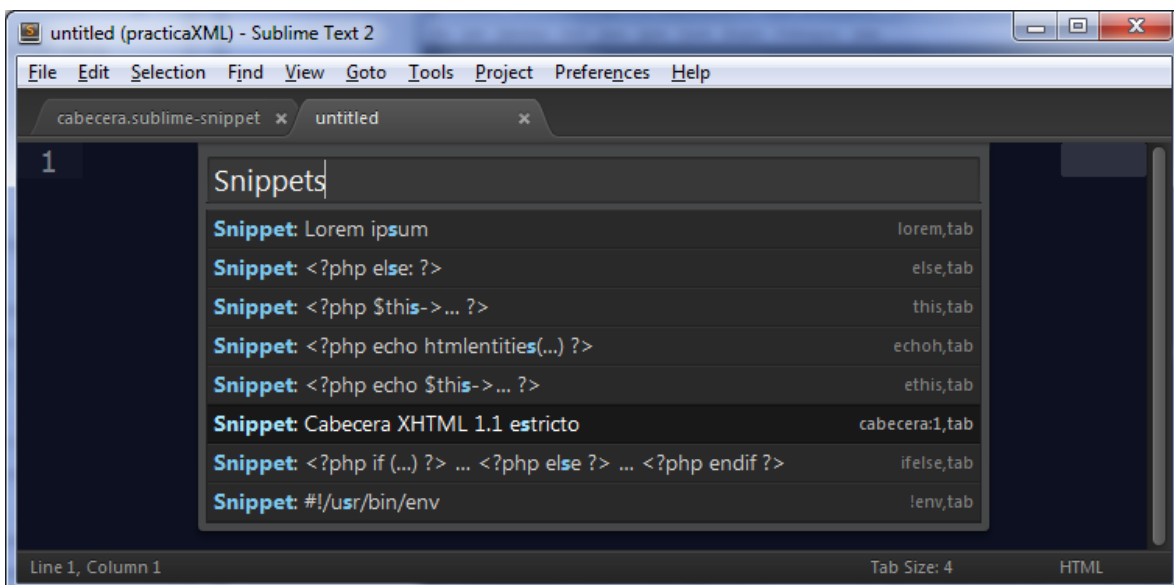


```

1 <snippet>
2   <content><![CDATA[
3     <?xml version="1.0" encoding="UTF-8"?>
4     <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
5       "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
6     <html version="-//W3C//DTD XHTML 1.1//EN"
7       xmlns="http://www.w3.org/1999/xhtml" xml:lang="es"
8       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
9       xsi:schemaLocation="http://www.w3.org/1999/xhtml
10      http://www.w3.org/MarkUp/SCHEMA/xhtml11.xsd">
11     <head>
12       <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
13       <meta name="lang" content="es" />
14       <title></title>
15     </head>
16     <body>
17     </body>
18   ]]>
19
20   </content>
21   <!-- Optional: Set a tabTrigger to define how to trigger the snippet -->
22   <tabTrigger>cabecera:1</tabTrigger>
23   <!-- Optional: Set a scope to limit where the snippet will trigger -->
24   <scope>text.html, source.php</scope>
25   <description>Cabecera XHTML 1.1 estricto</description>
26
27 </snippet>
28
  
```

En el ejemplo, dentro del apartado **CDATA** (en verde) se coloca el código común, el **tabTrigger** indica que el texto **cabecera:1** será el que invoque al snippet. Es decir cuando escribamos cabecera:1 y pulsemos tabulador, aparecerá el texto del Snippet.

En el apartado **description**, se coloca el texto que describirá al snippet cuando veamos la lista disponible mediante **Tools-Snippets** o en la paleta de comandos **Ctrl+Mayús+P** escribiendo **Snippets**.



```

1 Snippets
  Snippet: Lorem ipsum                                lorem,tab
  Snippet: <?php else: ?>                               else,tab
  Snippet: <?php $this->... ?>                           this,tab
  Snippet: <?php echo htmlentities(...) ?>              echoh,tab
  Snippet: <?php echo $this->... ?>                       ethis,tab
  Snippet: Cabecera XHTML 1.1 estricto                  cabecera:1,tab
  Snippet: <?php if (...) ?> ... <?php else ?> ... <?php endif ?> ifelse,tab
  Snippet: #!/usr/bin/env                               !env,tab
  
```

- (3) El paso tercero es guardar el Snippet con el nombre queramos en la carpeta **User**, pero con la extensión sublime-snippet
Por ejemplo **cabecera.sublime-snippet**
- (4) Usar el snippet desde la paleta o escribiendo el código **tabTrigger** (en el ejemplo (cabecera:1))

(5.4.2) introducir códigos de cursor

En el ejemplo anterior cuando se ejecuta el snippet, el cursor aparecerá tras el texto. Lo cual no es lo deseable, porque en ese ejemplo debería colocarse dentro del título (etiqueta **title**) y después pasar a escribir el cuerpo (etiqueta **body**) de la página web. Esto se hace añadiendo en el código del snippet los símbolos especiales **\$1**, **\$2**,...,

\$1 indica dónde quedará el cursor tras invocar al snippet, al pulsar el tabulador pasaremos a **\$2** y así sucesivamente. Es decir el snippet quedaría:

```
1 <snippet>
2   <content><![CDATA[
3     <?xml version="1.0" encoding="UTF-8"?>
4     <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
5       "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
6     <html version="-//W3C//DTD XHTML 1.1//EN"
7       xmlns="http://www.w3.org/1999/xhtml" xml:lang="es"
8       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
9       xsi:schemaLocation="http://www.w3.org/1999/xhtml
10      http://www.w3.org/MarkUp/SCHEMA/xhtml11.xsd">
11     <head>
12       <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
13       <meta name="lang" content="es" />
14       <title>$1</title>
15     </head>
16     <body>
17       $2
18     </body>
19   ]]>
20 ]>
21
22   </content>
23   <!-- Optional: Set a tabTrigger to define how to trigger the snippet -->
24   <tabTrigger>cabecera:1</tabTrigger>
25   <!-- Optional: Set a scope to limit where the snippet will trigger -->
26   <scope>text.html, text.xml, source.php</scope>
27   <description>Cabecera XHTML 1.1 estricto</description>
28 </snippet>
```

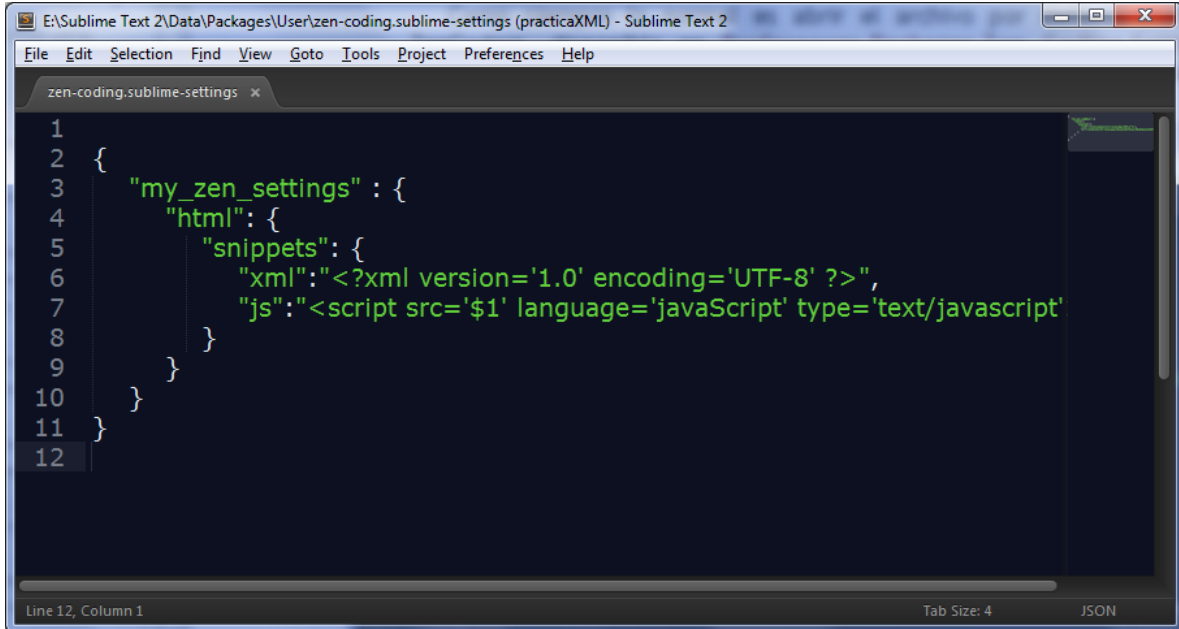
(5.4.3) snippets para ZenCoding

Los **snippets** que usamos para HTML y XML pueden no funcionar si hemos instalado el potente paquete **ZenCoding**. La razón es que ZenCoding siempre que pulsemos el tabulador va a intentar poner una etiqueta. Es decir si el código anterior tuviera como palabra clave **cabecera**, al escribir cabecera y pulsar tabulador, el resultado sería:

```
<cabecera>
</cabecera>
```

Una solución es usar **Ctrl+Espacio** para invocar al snippet que hemos creado (y esta solución es la más interesante) y no usar el tabulador. Pero también podemos usar el archivo de ajustes (**settings**) de ZenCoding y añadir un snippet.

Como siempre lo normal es abrir el archivo por defecto (**Settings-Default**) de ZenCoding, disponible en **Preferences-Packages-Zen Coding-Settings-Zen Settings Default** y echarle un vistazo para ver como funciona su formato. Después abriendo el **Zen Settings-User**, colocar el snippet. Ejemplo:



```
E:\Sublime Text 2\Data\Packages\User\zen-coding.sublime-settings (practicaXML) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help
zen-coding.sublime-settings x
1
2 {
3   "my_zen_settings" : {
4     "html": {
5       "snippets": {
6         "xml": "<?xml version='1.0' encoding='UTF-8' ?>",
7         "js": "<script src='$1' language='javaScript' type='text/javascript'
8       }
9     }
10  }
11 }
12

Line 12, Column 1
Tab Size: 4
JSON
```

(6) Builds

(6.1) qué es un build

En Sublime Text es posible corregir no sólo la sintaxis en línea, sino probar la ejecución del programa esté escrito en el lenguaje que sea.

Naturalmente eso implica tener instalado el compilador correspondiente (y poder acceder a él desde la variable **Path** del sistema, véase Apéndice 1). Es decir podemos escribir un programa en C usando Sublime Text y compilarlo sin salir de Sublime Text siempre y cuando esté disponible en línea de comandos el compilador de C y tengamos un archivo **build** apropiado en el que se indique qué compilador utilizar.

Sublime Text tiene varios **builds** predefinidos. Así si tenemos instalado el kit de desarrollo en Java podremos compilar directamente el código eligiendo **Tools-Build System-JavaC**, para indicar de qué modo compilaremos y después en cada programa Java podremos lanzar la aplicación mediante **Tools-Build** (o **ctrl+B**).

(6.2) archivos build

Para poder revisar programas propios que no disponga por defecto Sublime Text, tenemos que crear nuestros propios **builds**. Eso implica lo siguiente:

- (1) Elegir **Tools-Build System-New Build System**
- (2) Aparece una pantalla con un archivo JSON en el que se nos anima a modificar el contenido que pone **make** en el que pondremos el nombre del compilador que revisará el código con los parámetros que estemos oportunos

Un ejemplo de **build** complejo sería (extraído de http://sublimetext.info/docs/es/reference/build_systems.html):

```
{
  "cmd": ["ant"],
  "file_regex": "^ *\\[javac\\] (.+):([0-9]+):() (.*)$",
  "working_dir": "${project_path:${folder}}",
  "selector": "source.java",

  "windows":
  {
    "cmd": ["ant.bat"]
  }
}
```

En la misma página de la que se ha extraído el ejemplo se explican con más detalle los parámetros. Pero fundamentalmente hay dos principales:

- **cmd**. Que indica el nombre (podría incluir la ruta entera) al comando que compila el código.

- **file_regex.** Los compiladores muestran mensajes con los errores e información sobre la ejecución del archivo, este parámetro permite capturar esa información para que podamos navegar cómodamente mediante las teclas **F4** y **Mayús+F4**. Para ello hay que saber manejar adecuadamente expresiones regulares y eso supera la pretensión de este manual.
- **path.** Permite especificar una ruta a un directorio para modificar temporalmente la variable path del sistema y así acceder al compilador sin modificar realmente la variable path del sistema. Ejemplo:

Además para poder realizar expresiones complejas, podemos utilizar estas variables dentro de las expresiones que utilicemos:

Expresión	Significado
\$file	Extrae la ruta del archivo, por ejemplo <code>c:\proyectos\java\1\mainSis.java</code>
\$file_path	Extrae la ruta a la carpeta que contiene el archivo, por ejemplo: <code>c:\proyectos\java\1</code>
\$file_name	Extrae sólo el nombre (y extensión) del archivo, ejemplo: <code>mainSis.java</code>
\$file_base_name	Extrae sólo la extensión, en el ejemplo: <code>java</code>
\$file_extension	Extrae sólo el nombre, en el ejemplo: <code>mainSis</code>
\$packages	Ruta completa al directorio <code>Packages</code> de Sublime Text
\$project	Ruta completa al archivo de proyecto activo.
\$project_path	Ruta al directorio del proyecto activo.
\$project_name	Nombre y extensión del archivo de proyecto activo.
\$project_extension	Extensión del archivo de proyecto activo.
\$project_base_name	Nombre del archivo de proyecto activo.

Las de uso habitual son las coloreadas en azul en la tabla, especialmente **\$file**.

(6.3) creación de un sistema de build para PHP

Como ejemplo de **build**, en este apartado veremos como crear un **build** para PHP. El programa que compila un archivo PHP se llama precisamente **php** y es un ejecutable que se encuentra en la carpeta de instalación de PHP. Por ejemplo `c:\xampp\php` (la carpeta más habitual en la que se encuentra PHP cuando se instala **xampp** para Windows, entorno habitual de trabajo en modo local para PHP que incluye APACHE, PHP y MySQL entre otras aplicaciones).

Normalmente el acceso al programa PHP cuando programamos habitualmente en PHP estará en el path del sistema (para cambiar el path en Windows, véase el apéndice 1); pero es posible determinar un acceso al programa dentro del propio archivo build sin tocar el **path** del sistema. El **build** más sencillo sería:

```
{
  "cmd": ["php", "$file"],
  "path": "C:\\xampp\\php",
  "selector": "source.php"
}
```

Suponiendo que la ruta sea `c:\xampp\php` para llegar a PHP, la variable `$file` permite indicar el nombre del archivo actual como parámetro al compilador php. Las rutas se indican con doble barra (en el caso de Windows), porque si no fallarían. El apartado **selector** permite indicar que este sería el **build** apropiado para los archivos PHP cuando se indique **Automatic** en el apartado **Tools-Build System**.

Ese código se debe usar de esta manera:

- (1) Elegir **Tools-Build System-New Build System**, eso prepara un nuevo documento de configuración de **builds** en Sublime
- (2) Escribir el código
- (3) Guardar el archivo en la carpeta de usuario (**User** dentro de la carpeta **Packages** del programa Sublime Text) sin perder la extensión sublime-build (por ejemplo **php.sublime-build**)
- (4) Reiniciar el programa. En el apartado de menú **Tools-Build System** ahora aparecerá una nueva opción con el nombre que le dimos al archivo (por ejemplo **php**)
- (5) Una vez elegido nuestro Build System (**Tools-Build System-php** por ejemplo) pulsando **Ctrl+B** lanzaremos el compilador elegido.

Cuando compilemos aparecen los errores en un panel inferior. Si no hay errores, simplemente aparece el código HTML resultante. Los errores aparecen con un texto como:

Notice: Use of undefined constant x - assumed 'x' in C:\xampp\htdocs\pruebas\bucle1.php on line 12

En azul se ha coloreado el texto que siempre aparece en un error. A la izquierda del in aparece el tipo de error, a la derecha el archivo que provoca el error y detrás de **on line** aparece la línea en la que ocurre el error. Eso permite crear un código build que nos permite capturar los mensajes de error y recorrerlos de forma más cómoda con la tecla **F4**. Hay que hacer esta modificación al archivo build:

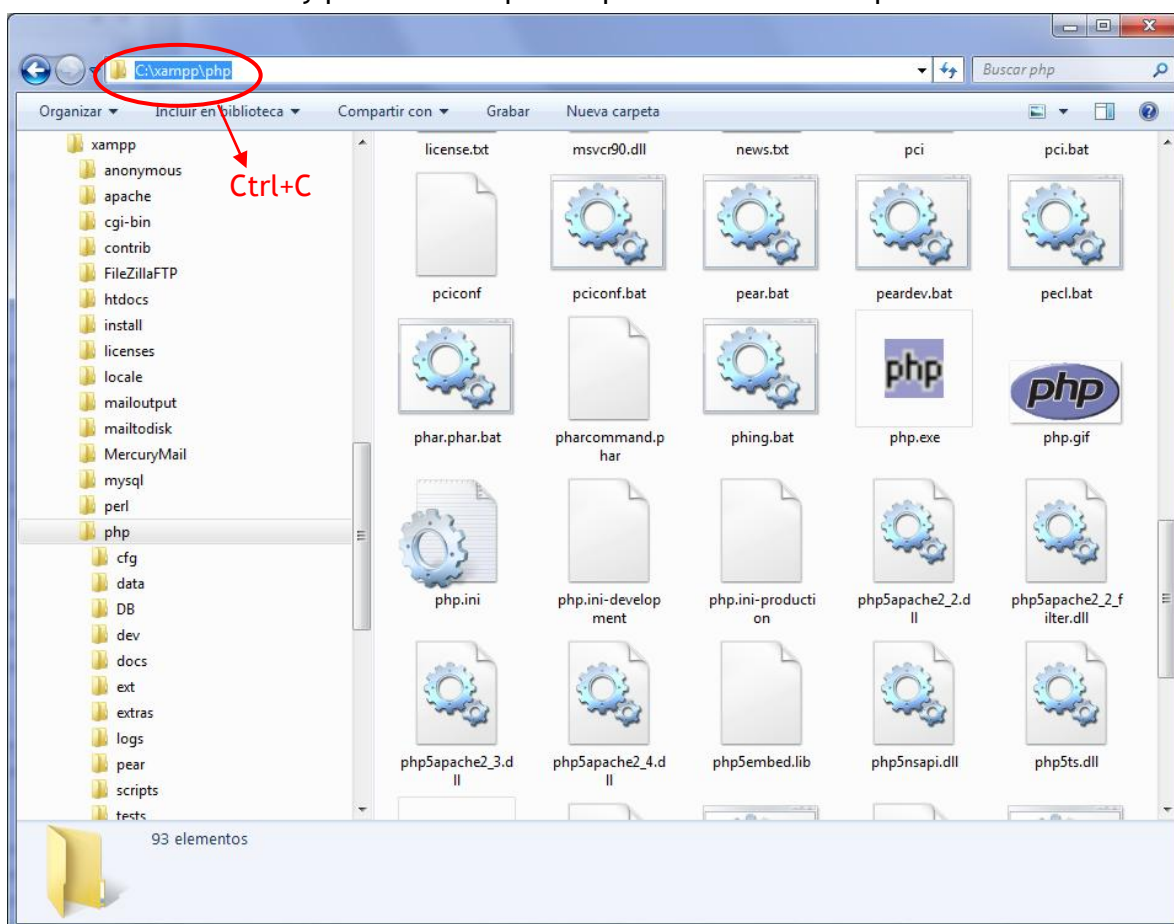
```
{
  "cmd": ["php", "$file"],
  "file_regex": ".*(?: in )(.*?) on line ([0-9]*)",
  "path": "C:\\xampp\\php",
  "selector": "source.php"
}
```

Gracias al apartado **file_regex** podremos navegar de error en error mediante la tecla **F4** (y retroceder con **Mayús+F4**).

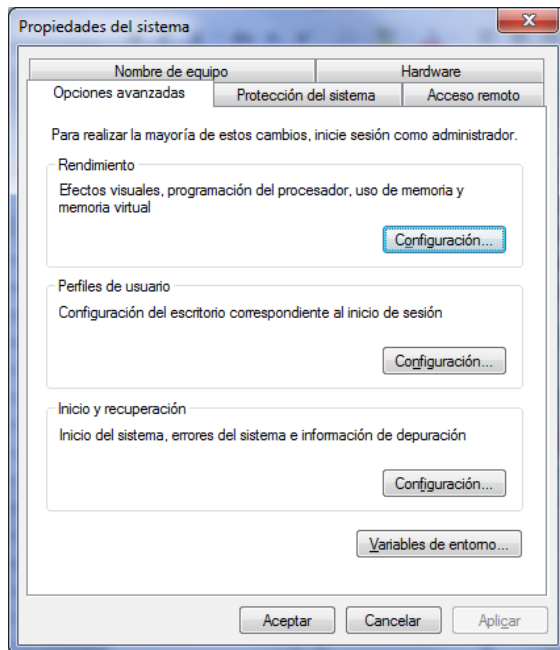
Apéndice 1: Modificación del Path del Sistema

Como se ha visto, en algunas partes de este manual, es necesario instalar programas que acompañan al propio Sublime Text a realizar algunas tareas. Eso significa que Sublime Text les llamará para ayudarle a realizar su labor y para que esa llamada funcione, la ruta en la que está el programa debe de estar metida en una variable del sistema conocida como PATH. Los pasos más visuales para cambiar la variable PATH en Windows son (vamos a tomar como ejemplo el programa php.exe que viene en la instalación de PHP):

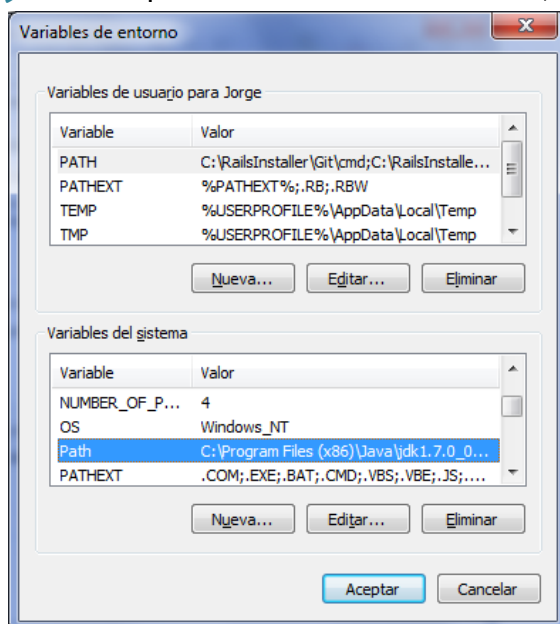
- (1) Ir a la carpeta en la que se encuentra el programa, hacer clic en la barra de direcciones y pulsar Ctrl+C para copiar la ruta de la carpeta:



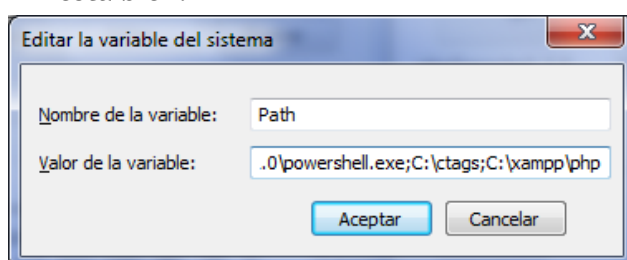
- (2) Pulsar el botón derecho sobre **Mi PC** (Windows 2000 y XP) o sobre Equipo (**Windows 7**) y elegir Propiedades
- (3) En Windows XP y similares podemos elegir Opciones avanzadas. En Windows 7 hay que elegir primero **Opciones Avanzadas del sistema** y luego **Opciones avanzadas**



- (4) Pulsar el botón **Variables de Entorno**
- (5) En el apartado **Variables de Sistema**, elegir **Path** y pulsar **Editar**.



- (6) Colocarse al final del texto ya escrito del **Path**, escribir ; (punto y coma) y pulsar **Ctrl+V** para pegar la ruta que hemos copiado. Hay que asegurarse de que la ruta está bien.



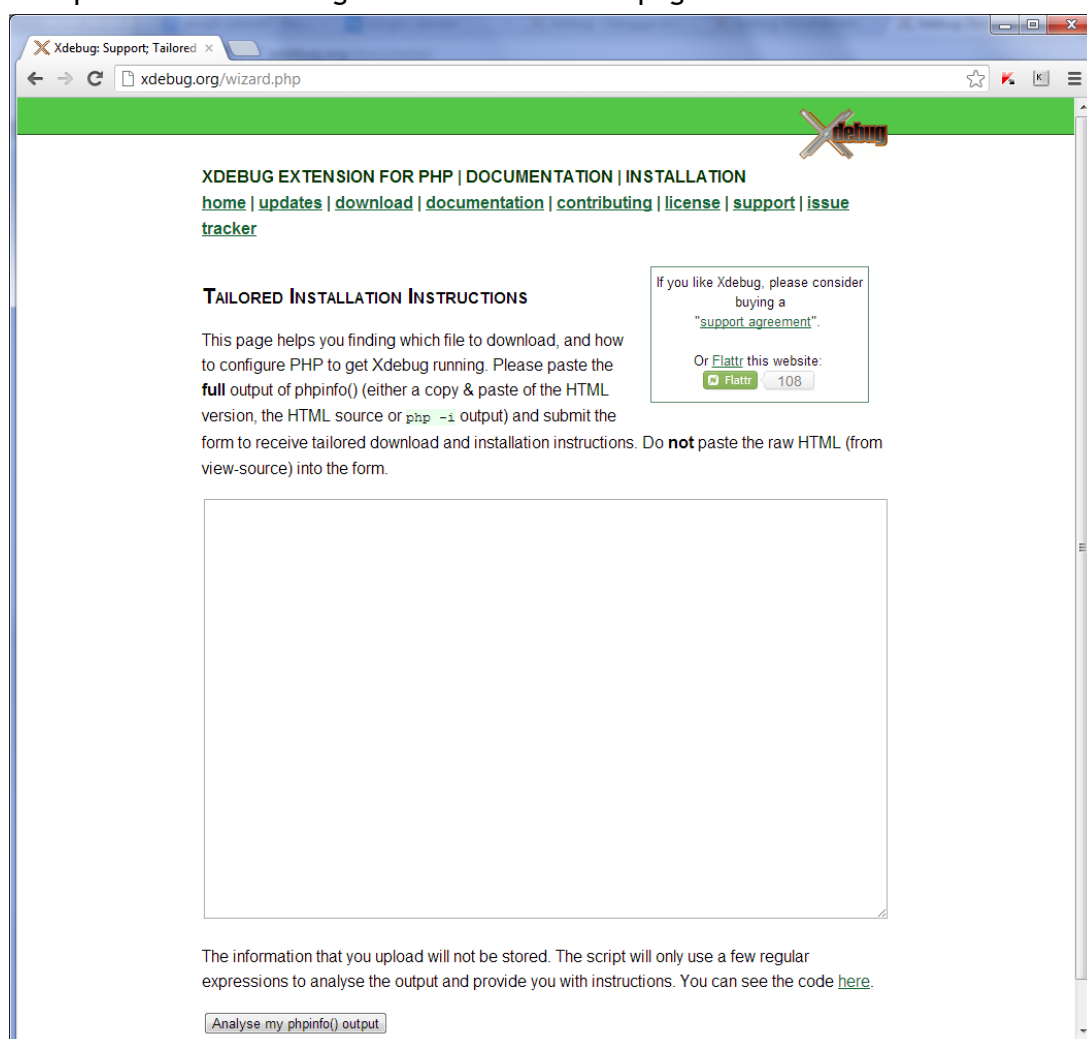
Apéndice 2: Uso de XDebug desde Sublime Text 2

introducción

XDebug es un depurador instalable para entornos PHP. Permite realizar trazas sobre el programa PHP. En el caso de la programación en lenguajes clásicos como C, C++, Pascal, Java, etc. la depuración era más sencilla ya que el entorno de depuración se ejecutaba directamente, sin embargo en los lenguajes de script de servidor como PHP (o JSP o ASP) resulta más complicado al ejecutarse en servidores de aplicaciones.

Por lo tanto los entornos de depuración se deben añadir como una extensión más. En el caso de XDebug (el depurador más popular para PHP), se trata de un entorno gratuito cuya descarga e información se hace desde www.xdebug.org. En el apartado [downloads](#) se puede descargar el depurador para versión de PHP que estemos usando.

No obstante lo más inteligente es ir a al apartado <http://xdebug.org/wizard.php> a través del cual la propia página nos dirá qué archivo descargar, donde colocarle e incluso qué archivo de configuración modificar. La página es esta:

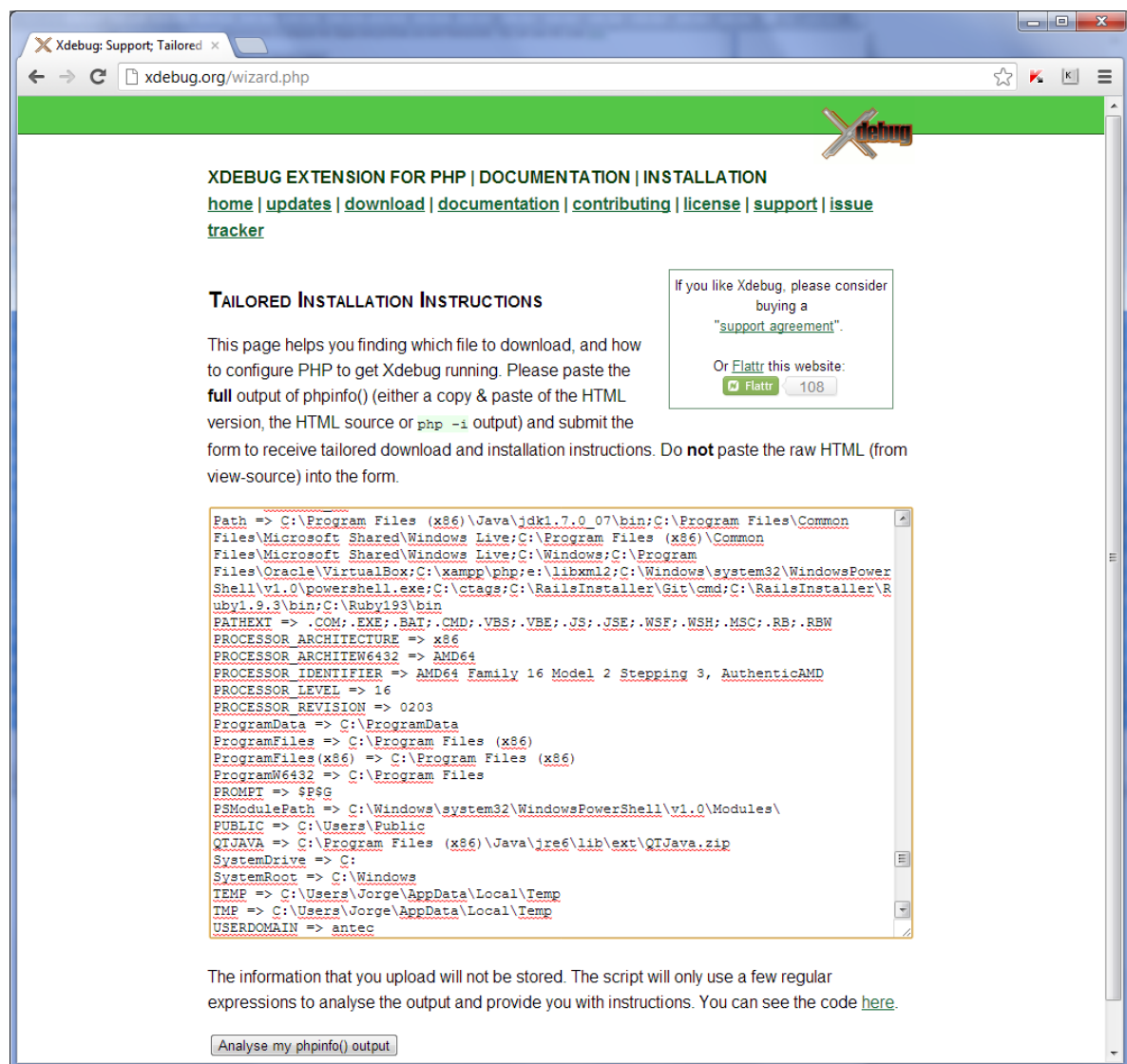


En esa página el gran cuadro central nos anima a pegar la información que muestre el comando **php -i**

Lógicamente para que ese comando funcione debemos tener instalado php y la ruta a su directorio estar metida en el **path** del sistema (véase Apéndice 1). Si esto está bien y desde línea de comandos hacemos:

```
php -i > phpinfo.txt
```

En el directorio en el que estemos crearemos un archivo de texto llamado **phpinfo.txt** en el que tendremos el texto que resulta de la ejecución de este comando. Ese es el texto que debemos meter en el cuadro (ese texto contiene toda la información disponible sobre la instalación de PHP a través de la función **phpinfo()**):



The screenshot shows the Xdebug Wizard website in a browser. The page has a green header with the Xdebug logo. Below the header, there are links for documentation and installation. The main section is titled "TAILORED INSTALLATION INSTRUCTIONS" and contains a paragraph explaining the purpose of the page. To the right of this section is a box asking users to consider buying a "support agreement" or to Flattr the website. Below the instructions, there is a large text area containing the output of the `php -i` command. The output lists various system and PHP configuration details, such as the path, processor architecture, and various file paths. At the bottom of the page, there is a button labeled "Analyse my phpinfo() output".

XDEBUG EXTENSION FOR PHP | DOCUMENTATION | INSTALLATION
[home](#) | [updates](#) | [download](#) | [documentation](#) | [contributing](#) | [license](#) | [support](#) | [issue tracker](#)

TAILORED INSTALLATION INSTRUCTIONS

This page helps you finding which file to download, and how to configure PHP to get Xdebug running. Please paste the **full** output of `phpinfo()` (either a copy & paste of the HTML version, the HTML source or `php -i` output) and submit the form to receive tailored download and installation instructions. Do **not** paste the raw HTML (from view-source) into the form.

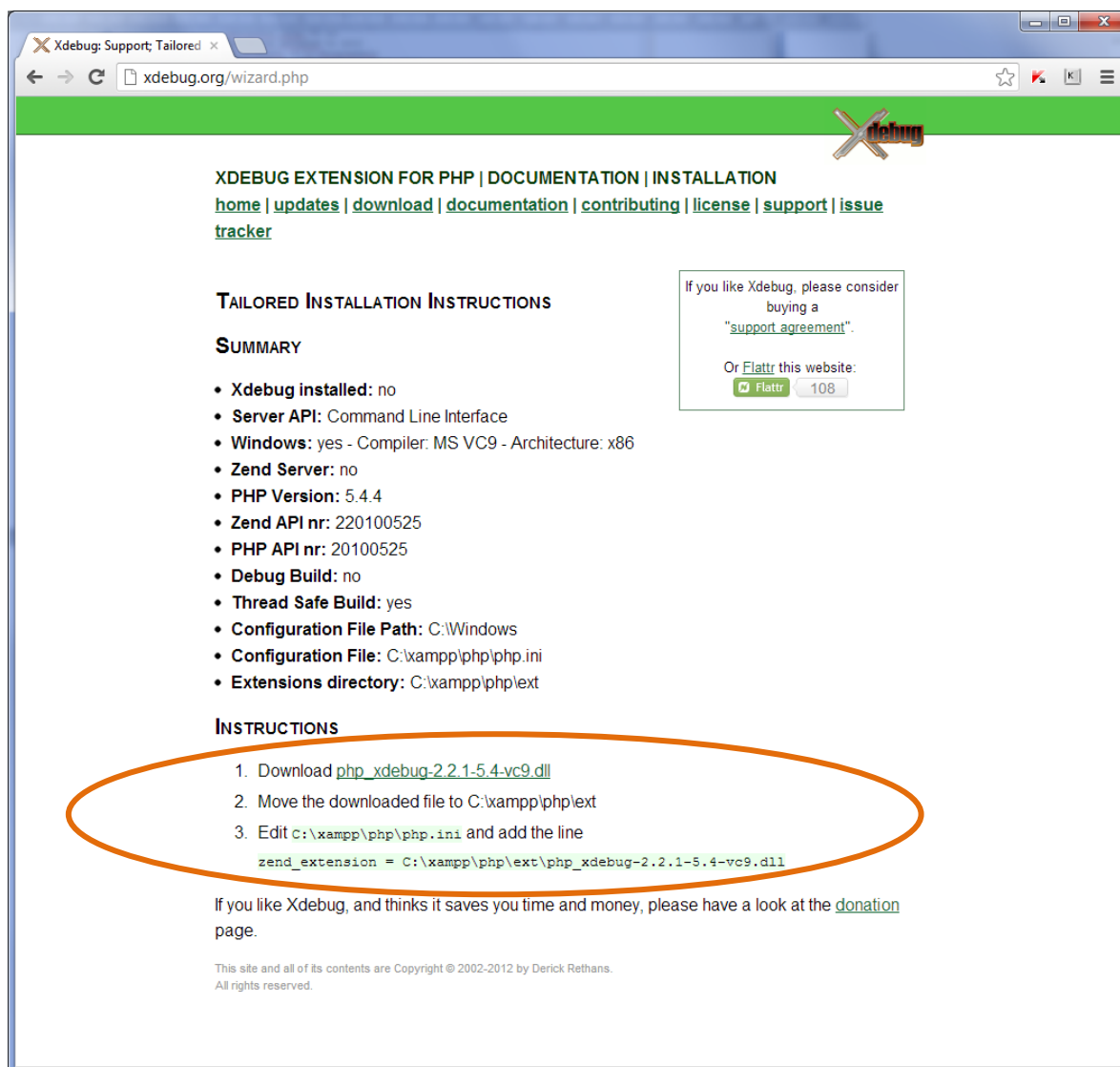
If you like Xdebug, please consider buying a "support agreement".
Or Flattr this website:
[Flattr](#) 108

```
Path => C:\Program Files (x86)\Java\jdk1.7.0_07\bin;C:\Program Files\Common
Files\Microsoft Shared\Windows Live;C:\Program Files (x86)\Common
Files\Microsoft Shared\Windows Live;C:\Windows;C:\Program
Files\Oracle\VirtualBox;C:\xampp\php;e:\libxml2;C:\Windows\system32\WindowsPower
Shell\v1.0\powershell.exe;C:\ctags;C:\RailsInstaller\Git\cmd;C:\RailsInstaller\R
uby1.9.3\bin;C:\Ruby193\bin
PATHEXT => .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC;.RB;.RBW
PROCESSOR_ARCHITECTURE => x86
PROCESSOR_ARCHITECTUREW6432 => AMD64
PROCESSOR_IDENTIFIER => AMD64 Family 16 Model 2 Stepping 3, AuthenticAMD
PROCESSOR_LEVEL => 16
PROCESSOR_REVISION => 0203
ProgramData => C:\ProgramData
ProgramFiles => C:\Program Files (x86)
ProgramFiles(x86) => C:\Program Files (x86)
ProgramW6432 => C:\Program Files
PROMPT => $P$G
FSModulePath => C:\Windows\system32\WindowsPowerShell\v1.0\Modules\
PUBLIC => C:\Users\Public
QTJAVA => C:\Program Files (x86)\Java\jre6\lib\ext\QTJava.zip
SystemDrive => C:
SystemRoot => C:\Windows
TEMP => C:\Users\Jorge\AppData\Local\Temp
TMP => C:\Users\Jorge\AppData\Local\Temp
USERDOMAIN => antec
```

The information that you upload will not be stored. The script will only use a few regular expressions to analyse the output and provide you with instructions. You can see the code [here](#).

Analyse my phpinfo() output

Ahora pulsamos en [Analyze my phpinfo\(\) output](#) y nos dirá lo que debemos descargar y modificar:



The screenshot shows the Xdebug Wizard page at xdebug.org/wizard.php. The page has a green header with the Xdebug logo. Below the header, there are links for documentation and installation. The main content area is titled "TAILORED INSTALLATION INSTRUCTIONS" and contains a "SUMMARY" section with a list of system details. To the right of the summary is a box asking for support. Below the summary is an "INSTRUCTIONS" section with three steps, the third of which is circled in orange. At the bottom, there is a donation link and a copyright notice.

XDEBUG EXTENSION FOR PHP | DOCUMENTATION | INSTALLATION
[home](#) | [updates](#) | [download](#) | [documentation](#) | [contributing](#) | [license](#) | [support](#) | [issue tracker](#)

TAILORED INSTALLATION INSTRUCTIONS

SUMMARY

- Xdebug installed: no
- Server API: Command Line Interface
- Windows: yes - Compiler: MS VC9 - Architecture: x86
- Zend Server: no
- PHP Version: 5.4.4
- Zend API nr: 220100525
- PHP API nr: 20100525
- Debug Build: no
- Thread Safe Build: yes
- Configuration File Path: C:\Windows
- Configuration File: C:\xampp\php\php.ini
- Extensions directory: C:\xampp\php\ext

If you like Xdebug, please consider buying a "support agreement".
Or [Flattr](#) this website: [Flattr](#) 108

INSTRUCTIONS

1. Download [php_xdebug-2.2.1-5.4-vc9.dll](#)
2. Move the downloaded file to C:\xampp\php\ext
3. Edit C:\xampp\php\php.ini and add the line
`zend_extension = C:\xampp\php\ext\php_xdebug-2.2.1-5.4-vc9.dll`

If you like Xdebug, and thinks it saves you time and money, please have a look at the [donation](#) page.

This site and all of its contents are Copyright © 2002-2012 by Derick Rethans.
All rights reserved.

Inicialmente se nos muestra la información obtenida y después se nos indican las instrucciones a realizar. En el ejemplo anterior se nos insta a descargar un archivo .dll, moverlo en la carpeta [c:\xampp\php\ext](#) y en el punto 3 que modifiquemos el archivo [php.ini](#) y añadamos la línea que se indica. Evidentemente esta información depende de cada instalación de PHP.

Si todo sale bien, al iniciar el servidor PHP (normalmente Apache con la extensión de PHP) todo irá bien y no habrá errores.

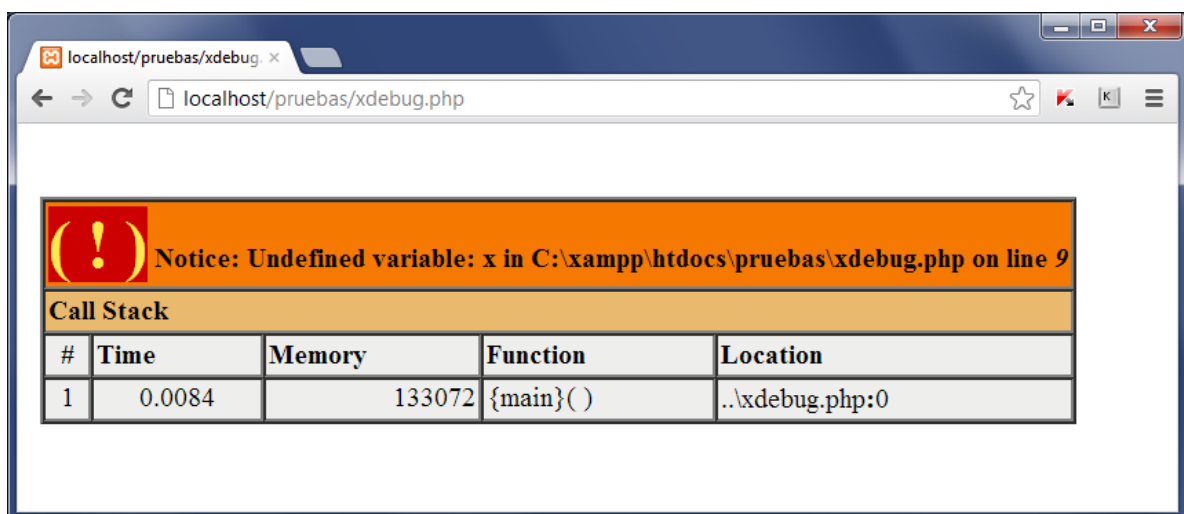
probar XDebug

Si XDebug está funcionando se hará cargo de los errores en PHP para darnos más información de la que muestra normalmente el servidor Web. Por ejemplo en el caso de que escribamos el programa:

```
<!DOCTYPE HTML>
<html lang="es-ES">
<head>
  <meta charset="utf-8" />
  <title>Prueba XDebug</title>
</head>
<body>
  <?php
      echo $x;

  ?>
</body>
</html>
```

La página resultante será:



La tabla que aparece nos indica que falta una variable y nos dice la línea en la que esto ocurre, además nos muestra datos de ejecución, función, tiempo, archivo original donde esta el error. Todo eso procede de XDebug, es información que no nos diría el servidor.

Además dispondremos de funciones propias del entorno XDebug, por ejemplo, este programa:

```
<?php
  echo xdebug_call_file()

?>
```

Escribe el nombre del archivo original PHP y sin XDebug instalado, no estaría esa función. Hay muchas más funciones, para saber más de ellas podemos leer la documentación oficial de <http://www.xdebug.org>

integrar XDebug en Sublime Text

instalación

Podemos no tener que esperar a ver los resultados de XDebug en la ventana del navegador donde ejecutamos el código PHP si integramos XDebug en el editor Sublime Text.

Para ello, una vez instalado XDebug en PHP, hay que añadir el paquete XDebug a través del **Package Control**, es decir:

- (1) Pulsar **Ctrl+Mayús+P** para abrir el editor de Comandos
- (2) Escribir simplemente **pack**, y elegir **Package Control-Install Package**
- (3) Escribir XDebug y elegirle de la lista para instalar el paquete
- (4) Reiniciar el editor.
- (5) Editar el archivo `php.ini` para asegurarnos que estas variables relacionadas con **XDebug** (**XDebug** normalmente tiene toda una sección en el `php.ini`) están configuradas de esta manera:

<code>xdebug.remote_enable=1</code>	<i>;Es la más importante y</i>
	<i>;normalmente la única a cambiar</i>
<code>xdebug.remote_host="localhost"</code>	<i>;Valor por defecto</i>
<code>xdebug.remote_port=9000</code>	<i>;Valor por defecto</i>
<code>xdebug.remote_handler="dbgp"</code>	<i>;Valor por defecto</i>

Por otro lado para trabajar con XDebug necesitamos usar proyectos y modificar los archivos de proyecto para indicar cuál es la dirección URL en la que se muestran las páginas PHP del proyecto.

Sin usar proyectos con PHP no es posible utilizar XDebug en Sublime Text, por otro lado es lógico usar PHP con proyectos, de modo que haya una carpeta en la que tenemos los archivos PHP que sea la asociada con el proyecto y cuya dirección URL correspondiente al servidor Web sea la que indicamos.

Por ejemplo, supongamos que hemos instalado **xampp** para Windows y usamos la carpeta **htdocs** para almacenar nuestras páginas PHP y esa carpeta la hemos abierto con **File-Open Folder** desde Sublime Text y hemos grabado un proyecto (con lo cual el proyecto se asocia a esa carpeta). Entonces debemos hacer lo siguiente:

- (1) Con el proyecto abierto, elegir **Project-Edit Project** esto nos abre el archivo del proyecto y podremos modificarle
- (2) Debemos añadir en la sección **settings**, la información sobre cuál es la URL sobre la que actúa XDebug. Por ejemplo:

```
{  
  "folders":  
  [  
    {  
      "path": "  
    }  
  ]  
}
```

```
{
    "path": "/C:/xampp/htdocs"
},

"settings": {
    "xdebug": { "url": "http://localhost/direcciónPHP" }
}
```

En el ejemplo se supone que el proyecto está asociado a la carpeta [C:\xampp\htdocs](#) y que la URL en el servidor PHP correspondiente a esa carpeta es simplemente <http://localhost>. Pero es muy importante tener en cuenta que la URL que se indica es la URL completa al archivo PHP que se quiere depurar con XDebug, por ejemplo: <http://localhost/practica1/ejercicio1.php>. Mientras que el apartado folders contiene simplemente el directorio (o directorios) incluidos en el proyecto.

uso

Una vez instalado la forma de probar y utilizar XDebug de forma integrada en XDebug es pulsar **Mayús+F8**, aparece una ventana desde la que podremos:

- Iniciar la depuración (**Start debugging**)
- Poner o quitar un punto de ruptura (**Add/remove breakpoint**). Un punto de ruptura aparece con un círculo blanco a la izquierda de la línea de código en la que se coloca y significa que, al lanzar la depuración, el programa se para ahí. En el navegador la página se queda en espera y si volvemos a Sublime Text observaremos en dos paneles la información completa de todas las variables y sus valores actuales (entre otras informaciones). La combinación **Ctrl+F8** hace la misma labor.

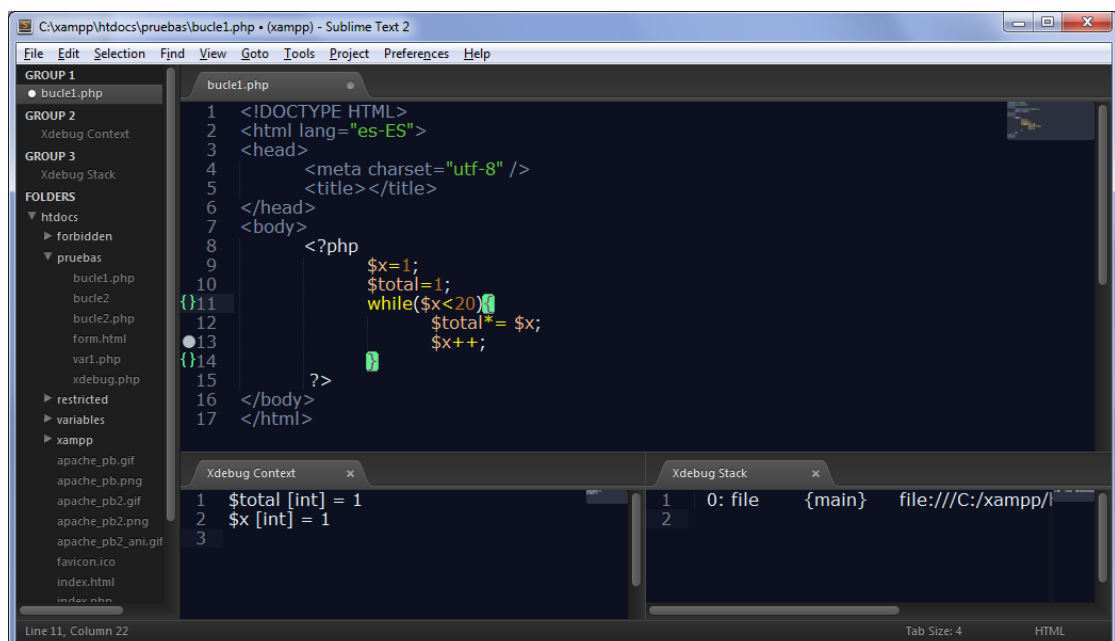


Ilustración 10, Sublime Text detenido tras un break point

- Detener la depuración (**Stop Debugging**), tras lo cual el programa ya no se detiene en los break points y finaliza su ejecución con normalidad. Se ocultan las ventanas de resultados.

Durante la depuración, la tecla F8 saca un panel que permite:

- Ejecutar (**Run**), es decir perseguir la ejecución hasta el siguiente punto de ruptura. Hace lo mismo la combinación **Ctrl+Mayus+F5**. Es la opción más habitual. En cada **Run** vamos al siguiente punto de ruptura para ir analizando como cambian las variables y así saber si el programa funciona de forma adecuada.
- Paso siguiente (**Step over**). Combinación **Ctrl+Mayus+F6**.
- Paso dentro (**Step into**). Hace lo mismo pero sigue línea a línea incluso dentro de las funciones PHP. Combinación **Ctrl+Mayus+F7**.
- Paso fuera (**Step out**). Sale de la función, si estamos dentro de una, para ir a la siguiente instrucción. **Ctrl+Mayus+F8**.